Efficient and Secure Post-Quantum Certificateless Signcryption for Internet of Medical Things

Shiyuan Xu, Xue Chen, Yu Guo, Siu-Ming Yiu, Shang Gao, and Bin Xiao

***Abstract*—Internet of Medical Things (IoMT) has gained significant research focus in both academic and medical institutions. Nevertheless, the sensitive data involved in IoMT raises concerns regarding user validation and data privacy. To address these concerns, certificateless signcryption (CLSC) has emerged as a promising solution, offering authenticity, confidentiality, and unforgeability. Unfortunately, most existing CLSC schemes are impractical for IoMT due to their heavy computational and storage requirements. Additionally, these schemes are vulnerable to quantum computing attacks. Therefore, research focusing on designing an efficient post-quantum CLSC scheme is still far-reaching. In this work, we propose PQ-CLSC, a novel post-quantum CLSC scheme that ensures quantum safety for IoMT. Our proposed design facilitates secure transmission of medical data between physicians and patients, effectively validating user legitimacy and minimizing the risk of private information leakage. To achieve this, we leverage lattice sampling algorithms and hash functions to generate the particial secret key and then employ the sign-then-encrypt method to obtain the ciphertext. We also formally and prove the security of our design, including indistinguishability against chosen-ciphertext attacks (IND-CCA2) and existential unforgeability against chosen-message attacks (EU-CMA) security. Finally, through comprehensive performance evaluation, our signcryption overhead is only 30%-55% compared to prior arts, while our computation overhead is just around 45% of other existing schemes. The evaluation results demonstrate that our solution is practical and efficient.**

***Index Terms*—Certificateless Signcryption, Internet of Medical Things, Information Security, Applied Cryptography.**

I. Introduction

THE Internet of Medical Things, a new concept emerging from the combination of medical sensor devices and the Internet of Things, providing patients with diverse and flexible treatment options [1], [2]. A traditional IoMT scenario consists of three types of entities, including patients, medical monitoring devices, and doctors [3], [4]. The medical monitoring device worn by the patient transmits data from various body indicators via the Internet to the hospital for storage. Doctors can access the patient's medical health data by accessing the database of patient records, and then use Artificial Intelligence algorithms to analyze the patient's data, point out possible conditions, provide remote treatment, prescribe potential medications, and make near real-time decisions for the patient. As the patient recovers, the doctor can also remotely ask the patient for advice and precautions to prevent the disease.

IoMT provides patients with convenient and reliable healthcare services, enabling them to prevent or treat diseases remotely and in a timely manner [5]. However, the data transmission mode of IoMT can be intercepted or even tampered with by an adversary during the communication process of the patient's medical information data, resulting in the leakage of a large amount of sensitive information such as the patient's personal data [6]. This could lead to doctors making incorrect diagnoses of patients' conditions. For example, if an adversary tampers with the data of medical monitoring devices and sends some worsened physical indicators to the hospital, the doctor may think that the patient's condition has worsened after the analysis and make a wrong diagnosis, thus affecting the patient's health [7]. Therefore, it is significant and challenging to transmit and protect medical data securely.

Numerous scholars have adopted digital signatures [8], [9] and public key encryption [10] to secure data transmissions between medical monitoring devices and users for user authentication and personal information protection. However, combining these cryptographic primitives in one scheme will significantly increase the computational and storage overhead, which is impractical for IoMT scenarios. Zheng [11] proposed an innovative primitive, namely signcryption, which can perform both encryption and signature operations. It not only satisfies the authenticity and confidentiality requirements but is also more effective than the 'sign then encrypt' or 'encrypt then sign' methodologies. The classical construction of CLSC protocols mainly includes two main categories, which are identity-based public key cryptography (IB-PKC) and public key infrastructure (PKI). Nevertheless, PKI-based CLSC schemes require a Certificate Authority (CA) to distribute a large number of certificates to users, resulting in complex management and high storage overhead. In addition, IB-PKC-based primitives face key escrow issues, where the key generation center (KGC) can arbitrarily decrypt the user's message and forge its signature, leading to security risks.

To overcome the problems mentioned above, Al-Riyami et al. [12] presented a certificate-less public key cryptography (CL-PKC) primitive. Unlike IBC, it introduces the semi-honest KGC with its master secret key, which is only responsible for generating part secret key of users. In 2008, Barbosa et al. [13] formalized the concept of CLSC based on bilinear pairing. In this protocol, a user's secret key consists of a secret key value of its own choice and a partial secret key. Since then, numerous novel CLSC schemes were proposed [14]–[17]. However, these schemes either require significant com-

Shiyuan Xu and Siu-Ming Yiu are with the Department of Computer Science, The University of Hong Kong, Pok Fu Lam, Hong Kong. (E-mail: syxu2@xs.hku.hk, smyiu@cs.hku.hk).

Shiyuan Xu and Yu Guo are with the School of Artificial Intelligence, Beijing Normal University, Beijing, China. (E-mail: yuguo@bnu.edu.cn).

Xue Chen, Shang Gao, and Bin Xiao are with the Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Hong Kong. (E-mail: xue-serena.chen@connect.polyu.hk, shang-jason.gao@polyu.edu.hk, b.xiao@polyu.edu.hk).

putational overhead or fail to provide data confidentiality in IoMT scenarios. Besides, most of the schemes are vulnerable to quantum attacks, which are insecure in the near future.

A. Motivation

IoMT provides patients with more reliable and convenient healthcare services, enabling them to receive treatment from doctors promptly. However, there are substantial security and privacy challenges when transmitting medical data in IoMT (e.g. It may be tampered with by malicious adversaries, and personal sensitive information of patients may be leaked). This phenomenon will result in a bottleneck for the development of IoMT. Therefore, how to protect the confidentiality and integrity of medical data while ensuring quantum-safety in IoMT is far-reaching.

In our design, we have to consider the practicality, efficiency, and security at a high level. To get around these, we aim to design a signcrption primitive that performs the roles of public key encryption and digital signature at the same time. In addition, we also need to involve lattice basis algorithms to resist quantum attacks. Moreover, to simplify the complexity of key management and deployment, a certificateless framework is promising since it avoids certificate management problems in public key infrastructure (PKI). As for security requirements, we should guarantee the confidentiality and unforgeability of the medical data, be able to resist quantum attacks and satisfy the properties of IND-CCA2 and UF-CMA.

B. Our Contribution

We summarize the fourfold contribution to this work below.

- We propose a novel efficient post-quantum certificateless signcryption scheme, namely PQ-CLSC, to achieve a secure medical data transmission between monitoring devices and users (patients and physicians) in the IoMT scenarios. It serves to validate the legitimacy of users and mitigate the risk of private information leakage. To the best of our knowledge, this is the first quantum-safe certificateless signcryption protocol for IoMT.
- The proposed scheme combines lattice-based certificateless signature and public key encryption into a single primitive. It offers several security advantages, including confidentiality, unforgeability, and authenticity of transmitted data under two type (Type I and Type II) attacks.
- Our scheme has been proven to satisfy IND-CCA2 and EU-CMA security in the random oracle model (ROM). Through rigorous security analysis, we demonstrate that the IND-CCA2 and EU-CMA security of our PQ-CLSC primitive can be reduced to the hardness of LWE and SIS, respectively. By conducting a comprehensive security comparison, our scheme successfully fulfills the desired properties of IND-CCA2, UF-CMA, and quantum resistance simultaneously, surpassing the capabilities of existing schemes.
- Through comprehensive experiments, we have determined that the signcryption and unsigncryption overheads of our PQ-CLSC scheme are 21.067 ms and 10.567 ms, respectively, resulting in a total computation overhead of 31.634 ms. Comparative analysis with other signcryption protocols [18]–[24] reveals that our PQ-CLSC scheme outperforms the overhead of all other lattice-based schemes. It is worth noting that our signcryption and unsigncryption overheads are only 0.30 to 0.55 times and 0.28 to 1.0 times compared to the other schemes, respectively. Our computation overhead is just 0.31 to 0.64 times of existing lattice-based signcryption schemes.

C. Technical Overview

Traditional lattice-based signcryption schemes adopt the encrypt-then-sign approach [25], [26]. However, it not works for the certificateless signcryption. In this way, our intuition is to leverage a hash function H_1 and $\mathsf{SampleD}$ technique into the partial secret key algorithm to compute $\mathbf{psk}_i$. Next, each user selects a secret value $\mathbf{s}_i$ and combines it with $\mathbf{psk}_i$ to obtain its secret key SK_i. In addition, most existing signcryption primitives utilized the $\mathsf{SamplePre}$ algorithm, resulting in tremendous computational burden [20]–[24]. Therefore, we avoid it for the sake of reducing the computational overhead of signcryption. Finally, we adopt a sign-then-encrypt methodology to construct the ciphertext.

In the **Setup** algorithm, we set a gadget matrix $\mathbf{G} := \mathbf{I}_n \otimes \mathbf{g}^\top$, $\mathbf{g}^\top = [1, 2, \cdots, 2^k], k = \lceil \log q \rceil - 1$ and utilize two universal hash functions $H_1 : \{0,1\}^* \times \{0,1\}^* \rightarrow \mathbb{Z}_q^n$ and $H_2 : \mathbb{Z}_q^{2n} \times \{0,1\}^* \rightarrow \{-1, 0, 1\}^k$. The KGC obtains the master public key and master secret key $(\mathbf{A}, \mathbf{T})$ by invoking $\mathsf{TrapGen}$ algorithm.

As for the **Partial secret key Extract** algorithm, a medical device or doctor will extract the partial secret key $\mathbf{psk}_i$ of user ID_i. The KGC calculates $\mathbf{u}_i$ by using a hash function $H_1(ID_i)$. The partial secret key $\mathbf{psk}_i \in \mathbb{Z}_q^\Theta$ is computed by $\mathsf{SampleD}$ algorithm.

In the **KeyGen** algorithm, a secret key SK_i consists of a random value $\mathbf{s}_i$ and a partial secret key $\mathbf{psk}_i$. We calculate the public key as $PK_i = (\mathbf{m}_i | \mathbf{M}_i^\top) \in \mathbb{Z}_q^{m \times (1+n)}$, where $\mathbf{M}_i$ is random selected and $\mathbf{m}_i = \mathbf{M}_i^\top \mathbf{x} + 2\mathbf{v}_i \bmod q$.

For the **Signcrypt** algorithm, we compute the ciphertext as $\mathbf{c} = (\mu_1 | \mu_2 | \mu)$. In particular, $\mu_1 = \mathbf{M}_U r + sig$, $\mu_2 = \mathbf{A}^\top \mathbf{w} + 2\mathbf{e}_2$ and $\mu = (2\mathbf{v}_U + m + \langle \mathbf{w}, H_1(ID_S, ID_U) \rangle + \langle \mathbf{m}_U, r \rangle) \bmod q$, which can be calculated by a signature sig, a user's ID, a user public key PK_U, a user secret key SK_S and random elements $r, \mathbf{w}, \mathbf{e}_2$.

D. Outline of this paper

Section II provides literature reviews to show the recent works. Then, we introduce the preliminary in Section III. After that, our system models, syntax, and security models are illustrated in Section IV. We elaborate on the proposed PQ-CLSC primitive in detail in Section V. In Sections VI and VII, we illustrate the security analysis as well as the comprehensive performance evaluation, respectively. Eventually, we conclude this paper.

II. Related Works

A. Signcryption

Signcryption primitives can play the roles of public key encryption and digital signature at the same time, ensuring

the confidentiality and integrity of data transmission. Its communication overhead is lower than that of the signing-then-encrypting scheme. It was originally proposed by Zheng et al. [11], which can perform signing and encryption algorithms in a single logical step. Malone-Lee et al. presented an Identity-based Signcryption scheme in 2002 [27], in which the public key may be any string. In 2005, Chen et al. formalized a more efficient Identity-based signcryption scheme in the random oracle model [28]. Subsequently, in 2008, Barbosa et al. [13] proposed the first certificateless signcryption scheme with bilinear pairing, providing forward secrecy and non-repudiation. Liu et al. then showed a novel secure certificateless signcryption scheme in a standard model [29], which is vulnerable to public key replacement attacks. The certificateless signcryption in paper [30] satisfies the requirements of unforgeability and confidentiality. In 2013, Yan et al. [31] gave a lattice-based signcryption scheme in a standard model resisting quantum computing attacks. Since then, numerous scholars have focused on designing certificateless signcryption primitives with quantum safety [32]–[34].

B. Internet of Medical Things

IoMT is a combination of medical sensor devices and the IoT [35]. It offers patients more convenient and reliable healthcare services, enabling them to seek treatment for their diseases more promptly. However, during the communication and transmission of healthcare data, it may be maliciously tampered with by adversaries, or patients' private data and personal data may be leaked [36]. Therefore, we require the help of cryptographic techniques to ensure the confidentiality and integrity of medical data transmission, the signcryption primitive is a promising candidate. In 2021, Zhang et al. [37] proposed the idea of utilizing the certificateless signcryption scheme to protect data in IoMT. The low computational and communication overhead of their scheme meets the demands of healthcare data transformation. However, at present, research on practical schemes for protecting healthcare data in the IoMT using signcryption primitive is scarce.

III. Preliminaries

This sector introduces several fundamental knowledge, including the notations utilized in this paper, definitions, and properties regarding Lattice, LWE hardness, SIS hardness, trapdoor algorithms, and leftover hash lemma. Table I explains the acronym and description used in this paper.

TABLE I
Nomenclature

Acronym	Description
λ	security parameter
q	prime number
B	error distribution parameter
s	Gaussian parameter
σ	discrete Gaussian distribution parameter
$\mathcal{D}$	discrete normal distribution
H_1, H_2	hash functions
pp	public parameter
(mpk, msk)	master public-secret key pair
ID_i	user's identity
S, U	signcrypt/unsigncrypt users set
$\{S, U\}$	all users set
ID_S, ID_U	signcrypt/unsigncrypt user
$\mathbf{psk}_i$	partial secret key of user ID_i
(PK_i, SK_i)	public-secret key pair of user ID_i
m	medical message
μ_1, μ_2, μ	ciphertext elements
sig, sig'	signature of ciphertext element μ_1
$\mathbf{c}$	final ciphertext of medical message m
$Q_{KG}, Q_{PSK}, Q_{PKR}, Q_{SV}$	query times
$\mathcal{O}_{H_1}^{list}$, $\mathcal{O}_{H_2}^{list}$, $\mathcal{O}_{PK}^{list}$	oracle lists
$\mathcal{B}_i$	simulation algorithms to solve problems
$\mathcal{A}_I / \mathcal{A}_{II}$	two-type adversaries
$\mathcal{C}$	challenger

*Definition 1 (**Lattice**):* [38] Given n linearly independent vectors $\mathbf{b_1}, \mathbf{b_2}, \cdots, \mathbf{b}_n \in \mathbb{R}^m$, an m-dimensional lattice Λ can be represented as

$$\Lambda = \Lambda(\mathbf{B}) = \{x_1 \cdot \mathbf{b_1} + x_2 \cdot \mathbf{b_2} + \cdots + x_n \cdot \mathbf{b}_n | x_i \in \mathbb{Z}\}. \quad (1)$$

We say $\mathbf{B} = \{\mathbf{b_1}, \mathbf{b_2}, \cdots, \mathbf{b}_n\} \subset \mathbb{R}^{m \times n}$ is a basis of Λ.

*Definition 2 (**Discrete Gaussian distribution**):* Given a positive parameter $\sigma \in \mathbb{R}^+$, a center $\mathbf{c} \in \mathbb{Z}^m$ and any $\mathbf{x} \in \mathbb{Z}^m$, we say that $\mathcal{D}_{\sigma,\mathbf{c}} = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{x})}{\rho_{\sigma,\mathbf{c}}(\Lambda)}$ for $\forall \mathbf{x} \in \Lambda$ is the discrete Gaussian distribution over Λ: $\rho_{\sigma,\mathbf{c}}(\mathbf{x}) = \exp(-\pi \frac{\|\mathbf{x}-\mathbf{c}\|^2}{\sigma^2})$, where $\mathbf{c}$ is a center and $\rho_{\sigma,\mathbf{c}}(\Lambda) = \sum_{\mathbf{x} \in \Lambda} \rho_{\sigma,\mathbf{c}}(\mathbf{x})$.

*Definition 3 (**Learning With Errors, LWE**):* [39] Given a positive integer n, $\alpha \in (0,1)$, a prime $q = q(n) > 2$, where $\alpha q > 2\sqrt{n}$, and a secret $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$, we define:

(1) LWE distribution: Uniformly select a matrix $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$, and sample $\mathbf{e} \leftarrow \Psi_\alpha^m$, output $(\mathbf{A}, \mathbf{A}^\top \mathbf{s} + \mathbf{e}) \in \mathbb{Z}_q^{n \times m} \times \mathbb{Z}_q^m$.

(2) Uniform distribution: Uniformly select a matrix $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$ and a vector $\mathbf{x} \xleftarrow{\$} \mathbb{Z}_q^m$, output $(\mathbf{A}, \mathbf{x}) \in \mathbb{Z}_q^{n \times m} \times \mathbb{Z}_q^m$.

Lemma 1: Given a vector $\mathbf{x} \leftarrow \mathcal{D}_{\mathbb{Z}^n, s}$ and the inequalities $\|\mathbf{x}\| \leq s\sqrt{n}$ and $|\mathbf{x}| \leq s\omega\sqrt{\log n}$ hold with overwhelming probability if $s \geq \omega\sqrt{\log n}$.

*Definition 4 (**Short Integer Solution, SIS**):* [38] Given a positive integer q, a random matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, m random vectors $\mathbf{a}_i \in \mathbb{Z}_q^n$, and a real number β ($q > \beta$), find a non-zero integer vector $\mathbf{z} \in \mathbb{Z}^m$ of norm $\|\mathbf{z}\| \leq \beta$ s.t.

$$\mathbf{Az} = \sum_i^m \mathbf{a}_i \cdot \mathbf{z}_i = 0 \in \mathbb{Z}_q^n. \quad (2)$$

Lemma 2: Informally speaking, we say $\mathcal{D}_{\sigma,0}^m$ abbreviated as $\mathcal{D}_\sigma^m$ when $\mathbf{c} = 0$. Given a vector $\mathbf{x} \leftarrow \mathcal{D}_\sigma^m$, it has $\|\mathbf{x}\| \leq 2\sigma\sqrt{m}$ with overwhelming probability. Given a real number $\lambda > 0$, and a vector $\mathbf{g} \in \mathbb{Z}^n$. We have

$$\Pr[\mathbf{x} \leftarrow \mathcal{D}_\sigma^m : \frac{\mathcal{D}_\sigma^m(\mathbf{x})}{\mathcal{D}_{\sigma,\mathbf{g}}^m(\mathbf{x})} < e^{\frac{1}{2\psi^2} + \frac{12}{\psi}}] > 1 - 2^{-100}, \quad (3)$$

where $\sigma = \psi(\|\mathbf{g}\|)$ and the probability distribution of $\mathcal{D}_\sigma^m$ is

$$\rho_{\sigma,\mathbf{c}}^m(\mathbf{x}) = e^{-(\mathbf{x} - \frac{\mathbf{c}^2}{2\sigma^2})} (2\pi\sigma^2)^{-\frac{m}{2}}. \quad (4)$$

*Theorem 1 (**TrapGen**):* [40] Given several parameters $n, m, q \in \mathbb{Z}$, this PPT algorithm publishes $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and $\mathbf{T_A} \in \mathbb{Z}_q^{m \times m}$, where $\mathbf{T_A}$ is a basis of $\Lambda_q^\perp(\mathbf{A})$ s.t.

$\{\mathbf{A} : (\mathbf{A}, \mathbf{T_A}) \leftarrow \textbf{TrapGen}(n, m, q)\}$ is statistically close to uniform and $\|\widetilde{\mathbf{T_A}}\| = \mathcal{O}(\sqrt{n \log q})$.

*Theorem 2 (**SampleD**):* [41] Assume $\mathbf{B}$ is a basis of a lattice Λ with dimension n. Given a parameter $s > 0$, and a center $\mathbf{c} \in \mathbb{R}^n$, this PPT algorithm outputs a vector $\mathbf{v} = \mathbf{v}_0$, where

$$\mathbf{v} - \mathbf{c} = \sum_{i \in [n]} (\mathbf{z}_i - \mathbf{c}_i) \cdot \mathbf{b}_i. \tag{5}$$

*Definition 5 (**Leftover Hash Lemma**):* [42] A simplified version of the leftover hash lemma includes two-universal functions $F = \{f : X \to Y\}$. Given two vectors $\mathbf{x}_1, \mathbf{x}_2 (\mathbf{x}_1 \neq \mathbf{x}_2)$, it always satisfies:

$$\Pr_{f \leftarrow F}(f(\mathbf{x}_1) = f(\mathbf{x}_2)) = \frac{1}{|Y|}. \tag{6}$$

Specifically, given a finite addition group $\mathbb{Z}_q^n$, any integer $m \geq 1$, and a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, the function $F = \{f_{\mathbf{A}} : \{0,1\}^m \to \mathbb{Z}_q^n; \mathbf{x} \mapsto f_{\mathbf{A}}(\mathbf{x}) = \mathbf{A}\mathbf{x}\}$ is two-universal.

IV. Problem Formulation

A. System Models

As elaborated in Fig. 1, an IoMT system normally involves five entities, namely, medical monitoring device (MMD), Gateway, key generation center (KGC), Physician, and medical cloud server (MCS).

- **MMD**: MMD is a device that is used to surveil various health indicators of a patient and is usually carried by the patient, e.g. Stethoscope holder, Sphygmomanometer, Continuous positive airway pressure, etc.
- **Gateway**: In an IoMT scenario, the gateway indicates a transfer center, linking the MMD data to the Gateway router through short-range radio transceivers. It acts as a communication bond between the MMD and a physician or MCS.
- **KGC**: KGC is the core infrastructure for public parameter and master public-secret key pairs generation. In addition, it also maintains to calculate the partial secret keys for MMD-embedded patients and physicians.
- **Physician**: Physician normally refers to the doctor-in-charge or rehabilitation therapist with the responsibility to communicate with the patients through Gateway and also exchange medical information from MCS. There exists a corresponding relationship between a signcrypt ciphertext stored in MCS and the private information of patients. Physicians can obtain the corresponding ciphertext from MCS according to patients' public information.
- **MCS**: MCS is a cloud server and it takes charge of medical data storage. After uploading the data to the MCS by MMD, a physician will diagnose the patient.

B. Formal Definitions of PQ-CLSC

A general PQ-CLSC scheme incorporates five PPT algorithms, **Setup**, **Partial secret key Extract**, **KeyGen**, **Signcrypt**, and **Unsigncrypt**. We specify the formal definitions of each algorithm below.

1) $(pp, (mpk, msk)) \leftarrow$ **Setup**(n, λ) : Given a system parameter n and a security parameter λ, this algorithm will be executed by KGC and output a public parameter pp and a master public-secret key pair (mpk, msk).
2) $\mathbf{psk}_i \leftarrow$ **Partial secret key Extract**(ID_i, pp) : Given a user with identity ID_i and a public parameter pp, this algorithm will return the user's partial secret key $\mathbf{psk}_i$.
3) $(PK_i, SK_i) \leftarrow$ **KeyGen**(ID_i, pp) : Given a user with identity ID_i and a public parameter pp, this algorithm calculates a secret key value $\mathbf{s}_i$ as intermediate and publishes a public-secret key pair (PK_i, SK_i) for ID_i.
4) $\mathbf{c} \leftarrow$ **Signcrypt**$(pp, m, ID_S, ID_U, SK_S, PK_U)$: Given a public parameter pp, a medical message m, a signcrypt user ID_S with its secret key SK_S, and an unsigncrypt user ID_U with its public key PK_U, this algorithm outputs a ciphertext $\mathbf{c}$.
5) m or $\perp \leftarrow$ **Unsigncrypt**$(pp, \mathbf{c}, ID_S, PK_S, ID_U, SK_U)$: Given a public parameter pp, a ciphertext $\mathbf{c}$, a signcrypt user ID_S with its public key PK_S, and an unsigncrypt user ID_U with its secret key SK_U, this algorithm publishes m or $\perp$ according to a judgment condition.

C. Security Models

There are two security prerequisites for a secure PQ-CLSC scheme, that is, confidentiality and unforgeability. Additionally, we need to consider two different types of malicious attackers (Type-I: $\mathcal{A}_I$ and Type-II: $\mathcal{A}_{II}$) interactive with one challenger $\mathcal{C}$ when designing the cryptographic primitive.

1) Security prerequisites
   - Confidentiality: A secure PQ-CLSC primitive requires to satisfy IND-CCA2, describing through several interactive games between $\mathcal{A}_I$ or $\mathcal{A}_{II}$ together with $\mathcal{C}$.
   - Unforgeability: A requirement for a secure PQ-CLSC primitive is to achieve EU-CMA, depicted through several interactive games between $\mathcal{A}_I$ or $\mathcal{A}_{II}$ together with $\mathcal{C}$. For further information about these security properties, please refer to references [43], [44].
2) Two types of adversaries
   - Type-I adversaries: A PPT adversary $\mathcal{A}_I$ has the ability to modify a user's public key PK_i but without learning any knowledge about the master secret key msk.
   - Type-II adversaries: A PPT adversary $\mathcal{A}_{II}$ masters the master secret key msk but doesn't have the ability to modify a user's public key PK_i.

V. Construction of Our Design

In this sector, we first illustrate the concrete construction of PQ-CLSC scheme, which can resist two types of adversary attacks and also satisfies IND-CCA2 and EU-CMA in a quantum-safe setting. Then, we give the parameter selections and the correctness analysis.

A. Initialization Phase

The KGC initializes the whole system by executing the **Setup** algorithm with the system parameter n and security parameter λ as input, then this algorithm processes the following procedures to generate a public parameter pp and a master public-secret key pair (mpk, msk).

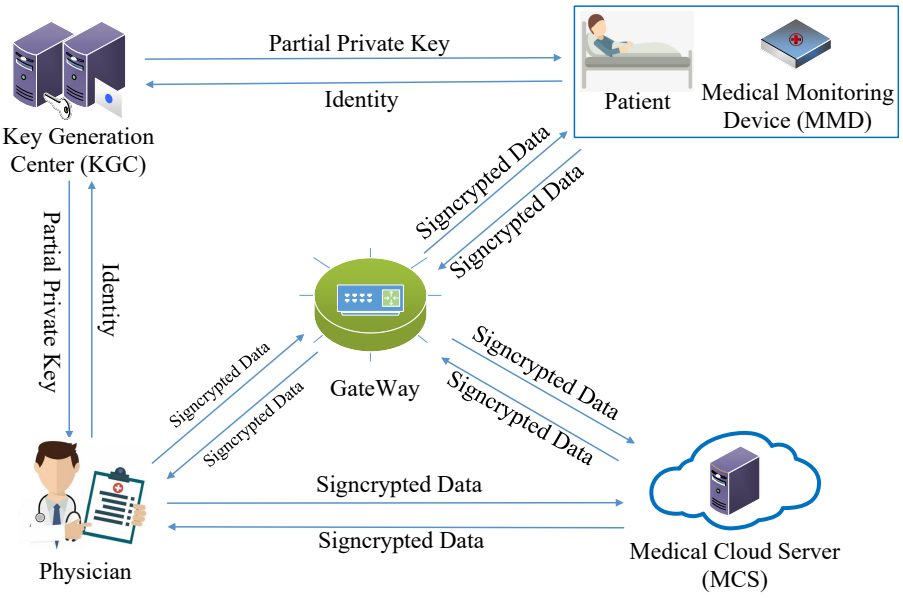

Fig. 1. The System Model of Our Proposed PQ-CLSC Scheme.

1) The KGC initially calls $q \leftarrow \mathsf{poly}(n)$, where q is a prime number. Then, KGC chooses $\alpha \xleftarrow{\$} \{0,1\}$ randomly.
2) The KGC also defines $\Theta = 2 \cdot n(\lceil \log q \rceil)$ and $m = O(n \log q)$ is a positive number. After that, it calculates the error distribution parameter $B = q \cdot \alpha \cdot \omega(\sqrt{\log n})$.
3) The KGC sets gadget matrix $\mathbf{G} := \mathbf{I}_n \otimes \mathbf{g}^\top$, $\mathbf{g}^\top = [1, 2, \cdots, 2^k], k = \lceil \mathsf{log} q \rceil - 1$.
4) The KGC selects two universal hash functions:

$$H_1 : \{0,1\}^* \times \{0,1\}^* \rightarrow \mathbb{Z}_q^n; \tag{7}$$

$$H_2 : \mathbb{Z}_q^{2n} \times \{0,1\}^* \rightarrow \{-1, 0, 1\}^k. \tag{8}$$

5) Moreover, the KGC executes the $\mathsf{TrapGen}(n, \Theta, q)$ algorithm to calculate $\mathbf{A} \in \mathbb{Z}_q^{n \times \Theta}$ and its basis $\mathbf{T} \in \mathbb{Z}_q^{\Theta \times \Theta}$.
6) In addition to this, KGC calculates a discrete Gaussian distribution $d = 4 \cdot \omega(\sqrt{\log n})$ and defines σ as the discrete Gaussian distribution parameter.
7) After that, the KGC defines master public key $mpk := \mathbf{A}$, master secret key $msk := \mathbf{T}$, and p as the lattice sampling parameter.
8) Ultimately, it returns a public parameter $pp := \{\mathbf{A}, \lambda, d, p, H_1, H_2\}$ and a master public-secret key pair (mpk, msk).

B. User Registration Phase

In the user registration phase, it contains two procedures to generate the public and secret keys for the patient. An medical entity (medical device or physician) firstly calculates and sends the partial secret key $\mathbf{psk}_i$ to the user with identity ID_i. Subsequently, the user calculates the public-secret key pair (PK_i, SK_i) by itself.

1) Generating the partial secret key: We hereby describe the first procedure to calculate the partial secret key. After takes a public parameter pp, and a user's identity ID_i as input, a medical entity extracts the partial secret key $\mathbf{psk}_i$ of user ID_i through the following **Partial secret key Extract** algorithm.

1) There are two user sets in the proposed scheme, namely the signcrypt users set and the unsigncrypt users set. We first define the signcrypt users set as $S := \{s_1, s_2, \cdots, s_\ell\}$, where ℓ is the total number of signcrypt users, $i \in [1, \ell]$, and $s_i \in \{0,1\}^*$. Then, we define the unsigncrypt users set $U := \{u_1, u_2, \cdots, u_\kappa\}$, where κ is the total number of unsigncrypt users, $i \in [1, \kappa]$, and $u_i \in \{0,1\}^*$.
2) The KGC calculates $\mathbf{u}_i = H_1(ID_i)$, where $ID_i \in \{S, U\} = \{s_1, s_2, \cdots, s_\ell, u_1, u_2, \cdots, u_\kappa\}$ denotes the general user.
3) The KGC parses $\bar{\mathbf{A}}$ through $\mathbf{A} = [\bar{\mathbf{A}} | \mathbf{G} - \bar{\mathbf{A}} | \mathbf{T}]$. After that, the KGC calls the $\mathsf{SampleD}(\bar{\mathbf{A}}, \mathbf{T}, \mathbf{u}_i, p)$ algorithm to obtain the partial secret key $\mathbf{psk}_i$ of user ID_i, where $\mathbf{psk}_i \in \mathbb{Z}_q^\Theta$.
4) Ultimately, the KGC sends $\mathbf{psk}_i$ to the user ID_i via a secure private channel.

2) Generating the public-secret key: Now, we move to the second to obtain the public key and secret key of the user. The user firstly takes a public parameter pp together with its identity ID_i as input to perform the **KeyGen** algorithm. After that, it calculates the public-secret key pair (PK_i, SK_i) corresponding to ID_i according to the following steps.

1) The user ID_i chooses a secret value $\mathbf{s}_i \xleftarrow{\$} D_{\mathbb{Z},q\alpha}^n \in \mathbb{Z}_q^n$ randomly and denotes its secret key as $SK_i = (\mathbf{s}_i, \mathbf{psk}_i) \in \mathbb{Z}^n \times \mathbb{Z}^\Theta$.
2) After that, the user ID_i chooses a matrix $\mathbf{M}_i \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$ and a vector $\mathbf{v}_i \xleftarrow{\$} D_{\mathbb{Z},q\alpha}^m \in \mathbb{Z}_q^m$ at random.
3) This algorithm calculates

$$\mathbf{m}_i = \mathbf{M}_i^\top \mathbf{x} + 2\mathbf{v}_i \bmod q \in \mathbb{Z}_q^m, \tag{9}$$

where vector $\mathbf{x} \leftarrow D_\sigma^n$ and $||\mathbf{x}|| \leq 2\sigma\sqrt{m}$.

4) Then, this algorithm calculates $PK_i = (\mathbf{m}_i|\mathbf{M}_i^\top) \in \mathbb{Z}_q^{m\times(1+n)}$ as a public key of user ID_i.

C. Ciphertext Generation Phase

In this phase, a signcrypt user ID_S takes a public parameter pp, a medical message m together with its secret key SK_S and the public key PK_U of an unsigncrypt user ID_U as input. Then, the signcrypt user performs the following **Signcrypt** algorithm to generate the ciphertext $\mathbf{c}$ and returns it to the unsigncrypt user.

1) Initially, a signcrypt user ID_S randomly chooses four vectors $\mathbf{r} \xleftarrow{\$} \{0,1\}^m, \mathbf{w} \xleftarrow{\$} D^n_{\mathbb{Z},q\alpha}, \mathbf{e}_1 \xleftarrow{\$} D_{\mathbb{Z},q\alpha}$, and $\mathbf{e}_2 \xleftarrow{\$} D_{\mathbb{Z},q\alpha}$.
2) Then, this user chooses $\epsilon_1 \xleftarrow{\$} D^l_\sigma \in \mathbb{Z}^l, \epsilon_2 \xleftarrow{\$} D^l_\sigma \in \mathbb{Z}^l$, and $\epsilon_3 \xleftarrow{\$} D^l_\sigma \in \mathbb{Z}^l$ randomly, and also defines a vector $\epsilon = \begin{bmatrix}\epsilon_1\\ \epsilon_2\\ \epsilon_3\end{bmatrix} \in \mathbb{Z}^{3l}$.
3) In addition, the user ID_S calculates a vector
$$\mathbf{g} = H_2(\begin{bmatrix}\mathbf{A} & \mathbf{A} & 0\\ 0 & \mathbf{M}_U & \mathbf{M}_U\end{bmatrix}\epsilon, m), \tag{10}$$
and a vector $\mathbf{t} = SK_S\mathbf{g} + \epsilon \in \mathbb{Z}^{3l}$.
4) Moreover, ID_S calculates a signature $sig' = \mathbf{t} + \mathbf{g}$,
$$sig = sig' \cdot (0, 0, \cdots, \lceil\frac{q}{2}\rceil)^\top \in \mathbb{Z}_q^n, \tag{11}$$
with probability $\mathsf{Prob} \geq \min(\frac{D^{3l}_\sigma(\mathbf{t})}{mD^{3l}_{\sigma,\omega}(\mathbf{t})}, 1)$.
5) Then, this signcrypt user ID_S calculates three ciphertext elements as below.
$$\mu_1 = \mathbf{M}_U\mathbf{r} + sig \in \mathbb{Z}_q^n, \tag{12}$$
$$\mu_2 = \mathbf{A}^\top\mathbf{w} + 2\mathbf{e}_2 \in \mathbb{Z}_q^\Theta, \tag{13}$$
$$\mu = (2\mathbf{v}_U + m + \langle\mathbf{w}, H_1(ID_S, ID_U)\rangle + \langle\mathbf{m}_U, \mathbf{r}\rangle) \bmod q. \tag{14}$$
6) Ultimately, ID_S defines and transmits the final ciphertext $\mathbf{c} = (\mu_1|\mu_2|\mu)$ to ID_U.

D. Ciphertext Decryption Phase

The unsigncrypt user ID_U takes a a public parameter pp, a ciphertext $\mathbf{c}$ together with its secret key SK_U and the public key PK_S of the signcrypt user ID_S as input. Then, the unsigncrypt user performs the following **Unsigncrypt** algorithm to decrypt the ciphertext $\mathbf{c}$ and thereby obtain the medical message m.

1) An unsigncrypt user ID_U first calculates
$$m = [\mu - \langle\mu_1, \mathbf{s}_U\rangle - \langle\mu_2, \mathbf{psk}_U\rangle]_q \bmod 2. \tag{15}$$
2) After that, the ID_U calculates
$$\mathbf{g}' = H_2(\begin{bmatrix}\mathbf{A} & \mathbf{A} & 0\\ 0 & \mathbf{M}_S & \mathbf{M}_S\end{bmatrix}\mathbf{t} - \begin{bmatrix}H_1(ID_S, ID_U)\\ PK_S\end{bmatrix}\mathbf{g}, m). \tag{16}$$
and verifies if the two following conditions hold:
$$||\mathbf{t}|| \leq 2\sigma\sqrt{3l} \text{ and } \mathbf{g}' \stackrel{?}{=} \mathbf{g}. \tag{17}$$
3) If the verification passes, ID_U will accept the medical message m; Otherwise, ID_U will output $\perp$, namely as the wrong medical message.

E. Parameters Setting

To enable the proposed scheme correctly and securely, we need to set several parameters as follows. For the security concern, we set $l \geq 5n\log q$. Then, considering the Gaussian parameter and discrete Gaussian distribution parameter, we need to make sure $s \geq \|\mathbf{T}\|\omega(\sqrt{\log n})$ and $\sigma \leq \alpha s\lambda\sqrt{6l}$. We also need to set the lattice sampling parameter $p = \sqrt{7(\mathsf{sv}(\mathbf{T})^2 + 1)}$, where $\mathsf{sv}(\mathbf{T})$ is the singular value of $\mathbf{T}$.

F. Correctness

We analyze the correctness of the proposed PQ-CLSC scheme through the following two steps.

1) Step 1: As for the unsigncrypt user ID_U, it has
$$m = [\mu - \langle\mu_1, \mathbf{s}_U\rangle - \langle\mu_2, \mathbf{psk}_U\rangle]_q \bmod 2, \tag{18}$$
when $|\mathbf{e}_1 + \mathbf{v}_S^\top - \mathbf{psk}_U^\top \cdot \mathbf{e}_2| < \frac{q}{4}$.
Proof □
$$\begin{aligned}&\mu - \langle\mu_1, \mathbf{s}_U\rangle - \langle\mu_2, \mathbf{psk}_U\rangle\\ &= (2\mathbf{v}_U + m + \langle\mathbf{w}, H_1(ID_S, ID_U)\rangle +\\ &\quad \langle\mathbf{m}_U, \mathbf{r}\rangle) - \langle\mu_1, \mathbf{s}_U\rangle - \langle\mu_2, \mathbf{psk}_U\rangle \bmod q\\ &= 2(\mathbf{e}_1 + \mathbf{v}_S^\top r - \mathbf{psk}_U^\top \cdot \mathbf{e}_2) + m \bmod q.\end{aligned} \tag{19}$$
If $(\mathbf{e}_1 + \mathbf{v}_S^\top\mathbf{r} - \mathbf{psk}_U^\top \cdot \mathbf{e}_2) < \frac{q}{4}$ holds, then we have $2(\mathbf{e}_1 + \mathbf{v}_S^\top\mathbf{r} - \mathbf{psk}_U^\top \cdot \mathbf{e}_2) < \frac{q}{2}$. Therefore, it has
$$m = [\mu - \langle\mu_1, \mathbf{s}_U\rangle - \langle\mu_2, \mathbf{psk}_U\rangle]_q \bmod 2 \tag{20}$$
2) Step 2: Our proposed **Signcrypt** algorithm is statistically indistinguishable from the distribution D^{3l}_σ according to the Lemma 1. In this way, we obtain $||\mathbf{t}|| \leq 2\sigma\sqrt{3l}$ with probability $\mathsf{Prob} \geq \min(\frac{D^{3l}_\sigma(\mathbf{t})}{mD^{3l}_{\sigma,\omega}(\mathbf{t})}, 1)$ and the equations as below.
$$\begin{aligned}&\begin{bmatrix}\mathbf{A} & \mathbf{A} & 0\\ 0 & \mathbf{M}_S & \mathbf{M}_S\end{bmatrix}\mathbf{t} - \begin{bmatrix}H_1(ID_S, ID_U)\\ PK_S\end{bmatrix}\mathbf{g}\\ &= \begin{bmatrix}\mathbf{A} & \mathbf{A} & 0\\ 0 & \mathbf{M}_S & \mathbf{M}_S\end{bmatrix}\mathbf{t} - \begin{bmatrix}H_1(ID_S, ID_U)\\ \mathbf{m}_S|\mathbf{M}_S^T\end{bmatrix}\mathbf{g}\\ &= \begin{bmatrix}\mathbf{A} & \mathbf{A} & 0\\ 0 & \mathbf{M}_U & \mathbf{M}_U\end{bmatrix}\mathbf{t} - \begin{bmatrix}\mathbf{A} & \mathbf{A} & 0\\ 0 & \mathbf{M}_U & \mathbf{M}_U\end{bmatrix}SK_U\mathbf{g}\\ &= \begin{bmatrix}\mathbf{A} & \mathbf{A} & 0\\ 0 & \mathbf{M}_U & \mathbf{M}_U\end{bmatrix}(\mathbf{t} - SK_U\mathbf{g})\\ &= \begin{bmatrix}\mathbf{A} & \mathbf{A} & 0\\ 0 & \mathbf{M}_U & \mathbf{M}_U\end{bmatrix}\boldsymbol{\epsilon}\end{aligned} \tag{21}$$

VI. Security Analysis

We analyze the security of the PQ-CLSC scheme with regard to confidentiality and unforgeability in this section. Our scheme satisfies IND-CCA2 and EU-CMA in a quantum setting under a random oracle model.

A. Confidentiality of PQ-CLSC

Theorem 3: If there exists a Type-I adversary $\mathcal{A}_I$ who has the ability to break IND-CCA2 of the proposed PQ-CLSC scheme with a non-negligible advantage Adv_{LWE} in probability-polynomial time, then there exists an algorithm $\mathcal{B}_1$ can solve the LWE hardness within $Q_{KG}+Q_{PSK}+Q_{PKR}+Q_{SV}$ query time, where $Q_{KG}, Q_{PSK}, Q_{PKR}, Q_{SV}$ means $\mathcal{A}_I$ can perform key generation query, partial secret key query, public key replace query, and secret value query, respectively.

Proof Suppose there exists a challenger $\mathcal{C}$ who can perform the algorithm $\mathcal{B}_1$. We finished the security analysis through three games as below.

Game 0: We simulate a real security game for an adversary $\mathcal{A}_I$ between a challenger $\mathcal{C}$. Given a system parameter n, $\mathcal{C}$ initially executes $(pp, (mpk, msk)) \leftarrow$ **Setup**(n, λ). Then, $\mathcal{C}$ sends pp to $\mathcal{A}_I$ and keeps the master secret key msk secret. In this way, $\mathcal{A}_I$ knows nothing about the msk. In addition, the challenger $\mathcal{C}$ maintains three lists $\mathcal{O}_{H_1}^{list}$, $\mathcal{O}_{H_2}^{list}$, and $\mathcal{O}_{PK}^{list}$ to record H_1 oracle, H_2 oracle, and public key oracle, respectively. These lists are initialized empty.

- Query 1 phase: The adversary $\mathcal{A}_I$ performs several queries and the challenger $\mathcal{C}$ will respond the corresponding messages to $\mathcal{A}_I$ as the following paragraphs.
  1) H_1 Query: After obtained the H_1 query of user ID_i from adversary $\mathcal{A}_I$, the challenger $\mathcal{C}$ looks up the $\mathcal{O}_{H_1}^{list}$ and returns the corresponding value $\mathbf{Hash}_i^1$ to $\mathcal{A}_I$ if this query $(ID_i, \mathbf{Hash}_i^1)$ has already in the $\mathcal{O}_{H_1}^{list}$; Otherwise, $\mathcal{C}$ selects $\mathbf{Hash}_i^1 \xleftarrow{\$} \mathbb{Z}_q^n$ randomly and inserts $(ID_i, \mathbf{Hash}_i^1)$ into the $\mathcal{O}_{H_1}^{list}$.
  2) H_2 Query: $\mathcal{A}_I$ firstly issues the H_2 query of medical message m, then $\mathcal{C}$ answers the corresponding value $\mathbf{Hash}^2$ to $\mathcal{A}_I$ if this query $(\mathbf{A}, \mathbf{M}_S, \epsilon, m)$ has already in the $\mathcal{O}_{H_2}^{list}$; Otherwise, $\mathcal{C}$ selects $\mathbf{Hash}^2 \xleftarrow{\$} \{-1, 0, 1\}^k$ and inserts $(\mathbf{A}, \mathbf{M}_S, \epsilon, m)$ into the $\mathcal{O}_{H_2}^{list}$.
  3) Public key request Query: After receiving the public key extract query of user ID_i from $\mathcal{A}_I$, $\mathcal{C}$ checks whether it exists $PK_i \in \mathcal{O}_{PK}^{list}$. If it holds, $\mathcal{C}$ will give PK_i to $\mathcal{A}_I$; Otherwise, $\mathcal{C}$ will calculate and give $PK_i \leftarrow (\mathbf{m}_i | \mathbf{M}_i^\top) \in \mathbb{Z}_q^{m \times (1+n)}$ to $\mathcal{A}_I$, and also insert $(\mathbf{ID}_i, *, *, \mathbf{s}_i, \mathbf{m}_i, \mathbf{M}_i, \mathbf{v}_i)$ into the $\mathcal{O}_{PK}^{list}$.
  4) Partial secret key extract Query: After obtaining the partial secret key extract query of user ID_i from adversary $\mathcal{A}_I$, the challenger $\mathcal{C}$ executes $\mathbf{psk}_i \leftarrow$ **Partial secret key Extract**(ID_i, pp). After that, $\mathcal{C}$ sends the $\mathbf{psk}_i$ to $\mathcal{A}_I$ and then inserts $(ID_i, *, \mathbf{psk}_i)$ into the $\mathcal{O}_{PK}^{list}$.
  5) Public key replace Query: $\mathcal{A}_I$ selects and sends a novel public key PK_i' to $\mathcal{C}$. Then, $\mathcal{C}$ retrieves the public key oracle list $\mathcal{O}_{PK}^{list}$ and updates PK_i to PK_i' corresponding to the ID_i.
  6) Secret key extract Query: After getting a query of user ID_i from adversary $\mathcal{A}_I$, $\mathcal{C}$ checks whether $(ID_i, PK_i) \in \mathcal{O}_{PK}^{list}$. If it holds and PK_i has not been replaced, $\mathcal{C}$ executes $SK_i \leftarrow$ **KeyGen**(ID_i, pp) for ID_i. Then, $\mathcal{C}$ gives the SK_i to $\mathcal{A}_I$ and inserts (ID_i, SK_i) into the $\mathcal{O}_{PK}^{list}$. Otherwise, $\mathcal{C}$ aborts it.
  7) Signcrypt Query: To begin with, $\mathcal{C}$ chooses $S' \xleftarrow{\$} \{1, 2, \cdots, \lceil q \rceil\}$ at random. In addition, $\mathcal{A}_I$ chooses ID_S, ID_U, and m as the signcrypt user's identity, unsigncrypt user's identity, and a medical message, respectively. When acquiring a signcrypt query from $\mathcal{A}_I$, $\mathcal{C}$ verifies $ID_S \stackrel{?}{=} ID_{S'}$. If it holds, $\mathcal{C}$ processes and sends $\mathbf{c} \leftarrow$ **Signcrypt**$(pp, m, ID_S, ID_U, SK_S, PK_U)$ to $\mathcal{A}_I$. Otherwise, $\mathcal{C}$ performs the following operations:
     - $\mathcal{C}$ initially selects $\epsilon \xleftarrow{\$} \mathbb{Z}^{3l}$ and $\mathbf{M}_U \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$.
     - Furthermore, $\mathcal{C}$ calculates
     $$\mathbf{g} = H_2(\begin{bmatrix} \mathbf{A} & \mathbf{A} & 0 \\ 0 & \mathbf{M}_U & \mathbf{M}_U \end{bmatrix} \epsilon, m), \tag{22}$$
     and inserts $(\mathbf{A}, \mathbf{M}_U, \epsilon, \mathbf{g})$ into $\mathcal{O}_{H_2}^{list}$.
     - Moreover, $\mathcal{C}$ calculates the signature $sig' = \mathbf{t} + \mathbf{g} = SK_S \mathbf{g} + \epsilon + \mathbf{g}$, $\mu_1 = \mathbf{M}_U r + sig$, $\mu_2 = \mathbf{A}^\top \mathbf{w} + 2\mathbf{e}_2$, and $\mu = (2\mathbf{v}_U + m + \langle \mathbf{w}, H_1(ID_S, ID_U) \rangle + \langle \mathbf{m}_U, \mathbf{r} \rangle) \bmod q$ accordingly.
     - Ultimately, $\mathcal{C}$ calculates the ciphertext $\mathbf{c} = (\mu_1 | \mu_2 | \mu)$ and sends it to $\mathcal{A}_I$.
  8) Unsigncrypt Query: At the beginning, $\mathcal{A}_I$ selects ID_S, ID_U, and m as the signcrypt user's identity, unsigncrypt user's identity, and a medical message, respectively. When acquiring a signcrypt query from $\mathcal{A}_I$, $\mathcal{C}$ verifies $ID_S \stackrel{?}{=} ID_{S'}$, where $S' \xleftarrow{\$} \{1, 2, \cdots, \lceil q \rceil\}$. If it holds, $\mathcal{C}$ calls and returns m or $\perp \leftarrow$ **Unsigncrypt**$(pp, \mathbf{c}, ID_S, PK_S, ID_U, SK_U)$ to $\mathcal{A}_I$; Otherwise, $\mathcal{C}$ manipulates the following steps: (1) $\mathcal{C}$ firstly calculates $\mathbf{g}'$ as
     $$H_2(\begin{bmatrix} \mathbf{A} & \mathbf{A} & 0 \\ 0 & \mathbf{M}_S & \mathbf{M}_S \end{bmatrix} \mathbf{t} - \begin{bmatrix} H_1(ID_S, ID_U) \\ PK_S \end{bmatrix} \mathbf{g}, m). \tag{23}$$
     (2) After that, $\mathcal{C}$ calculates
     $$m = [\mu - \langle \mu_1, \mathbf{s}_U \rangle - \langle \mu_2, \mathbf{psk}_U \rangle]_q \bmod 2 \tag{24}$$
     (3) Finally, $\mathcal{C}$ verifies $\mathbf{g}' \stackrel{?}{=} \mathbf{g}$. If the equation holds, $\mathcal{C}$ publishes m to $\mathcal{A}_I$; Otherwise, $\mathcal{C}$ publishes $\perp$ to $\mathcal{A}_I$.
- Challenge phase: The adversary $\mathcal{A}_I$ chooses two different medical messages with same length (m_0, m_1) corresponding to the signcrypt user ID_S^* and unsigncrypt user ID_U^*. In the current query, $\mathcal{A}_I$ is not permitted to obtain SK_i of ID_U^*. At this time, we suppose that $\mathcal{C}$ has finished the H_1 Query, Public key request Query, Partial secret key extract Query, and Secret key extract Query. $\mathcal{C}$ responds to the challenge query according to the following methods.
  1) If $ID_U^* \neq ID_S^{'}$, $\mathcal{C}$ will fail this game.
  2) Otherwise, $\mathcal{C}$ defines a vector $\epsilon^* = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{bmatrix} \in \mathbb{Z}^{3l}$, where l is a positive number s.t. $l \geq 5n \log q$ and then selects $b \xleftarrow{\$} \{0, 1\}$ at random. After that, $\mathcal{C}$ computes several equations as below.
     $$\mathbf{g}^* = H_2(\begin{bmatrix} \mathbf{A} & \mathbf{A} & 0 \\ 0 & \mathbf{M}_U & \mathbf{M}_U \end{bmatrix} \epsilon^*, m_b), \tag{25}$$
     $$sig'^* = \mathbf{t}^* + \mathbf{g}^* = SK_S \mathbf{g}^* + \epsilon^* + \mathbf{g}^*, \tag{26}$$

$$sig^* = sig'^* \cdot (0, 0, \cdots, \lceil \frac{q}{2} \rceil)^\top, \tag{27}$$

$$\mu_1^* = \mathbf{M}_U \mathbf{r} + sig^* \tag{28}$$

$$\mu_2^* = \mathbf{A}^\top \mathbf{w} + 2\mathbf{e}_2^* \tag{29}$$

$$\begin{aligned} \mu^* =& (2\mathbf{v}_U + m_b + \langle \mathbf{w}, H_1(ID_S^*, ID_U^*) \rangle + \\ & \langle \mathbf{m}_U, \mathbf{r} \rangle) \bmod q. \end{aligned} \tag{30}$$

Ultimately, $\mathcal{C}$ sends the challenge ciphertext $\mathbf{c}^* = (\mu_1^* | \mu_2^* | \mu^*)$ to $\mathcal{A}_I$ and inserts $\mathbf{c}^*$ to $\mathcal{O}_{H_2}^{list}$.

This is the end of Query 1.

- Query 2 phase: In this query, the adversary $\mathcal{A}_I$ can access almost exactly the same queries as in Query 1 except that $\mathcal{A}_I$ is forbidden to access the Partial Secret key extract Query and Secret key extract Query with inputting (ID_i^*, pp) and (ID_S^*, ID_U^*), respectively. Besides, $\mathcal{A}_I$ is also forbidden to access the Unsigncrypt Query by inputting $\mathbf{c}^*$.
- Guess phase: Finally, $\mathcal{A}_I$ outputs a guess b'. Then, $\mathcal{C}$ verifies if $b' \stackrel{?}{=} b$. If it holds, $\mathcal{C}$ will output a solution of the LWE hardness; Otherwise, $\mathcal{C}$ will output $\perp$.

We define $Adv_{\mathcal{A}_I}^{\widehat{\textbf{Game 0}}}(\lambda)$ as the advantage of $\mathcal{A}_I$ wins the $\widehat{\textbf{Game 0}}$.

$\widehat{\textbf{Game 1}}$: This game is identical to $\widehat{\textbf{Game 0}}$, except for $\mathbf{psk}_i$ in the partial secret key extract Query. Concretely, $\mathcal{C}$ chooses $\mathbf{psk}_i \xleftarrow{\$} D_{\mathbb{Z}^\Theta, p \cdot \omega(\log n)}$ randomly and then computes $\mathbf{u}_i = \mathbf{A}\mathbf{psk}_i$. If $\mathbf{u}_i \notin \mathcal{O}_{H_1}^{list}$, $\mathcal{C}$ defines $H_1(ID_i) = \mathbf{u}_i$; If $\mathbf{u}_i \in \mathcal{O}_{H_1}^{list}$, $\mathcal{C}$ recalculates $\mathbf{psk}_i \leftarrow$ **Partial secret key Extract**(ID_i, pp).

We define $Adv_{\mathcal{A}_I}^{\widehat{\textbf{Game 1}}}(\lambda)$ as the advantage of $\mathcal{A}_I$ wins the $\widehat{\textbf{Game 1}}$.

As for $\mathcal{A}_I$, $\widehat{\textbf{Game 1}}$ and $\widehat{\textbf{Game 0}}$ are statistically indistinguishable due to the property of the lattice sampling algorithm. Consequently, we obtain:

$$|Adv_{\mathcal{A}_I}^{\widehat{\textbf{Game 1}}}(\lambda) - Adv_{\mathcal{A}_I}^{\widehat{\textbf{Game 0}}}(\lambda)| \leq \textbf{negl}(\lambda). \tag{31}$$

$\widehat{\textbf{Game 2}}$: This game is identical to $\widehat{\textbf{Game 1}}$, except changing the calculation method of master public key $mpk := \mathbf{A}$. More concretely, we specify the process as follows.

- Setup phase: To begin with, $\mathcal{C}$ executes $pp \leftarrow \mathbf{Setup}(n, \lambda)$ to achieve the randomness for $\mathbf{A}$. Then, $\mathcal{C}$ sends the public parameter pp to $\mathcal{A}_I$.
- Query phase: In $\widehat{\textbf{Game 2}}$, $\mathcal{A}_I$ can nearly access the same queries as in the $\widehat{\textbf{Game 0}}$, excepting two queries.
  1) Partial secret key extract Query: After obtaining the partial secret key extract query of user ID_i from $\mathcal{A}_I$, $\mathcal{C}$ executes $\mathbf{psk}_i \leftarrow$ **Partial secret key Extract**(ID_i, pp) and also obtains $\mathbf{u}_i = H_1(ID_i)$. After that, $\mathcal{C}$ sends the $\mathbf{psk}_i$ to $\mathcal{A}_I$ and then inserts $(ID_i, \mathbf{u}_i, \mathbf{psk}_i)$ into the $\mathcal{O}_{PK}^{list}$.
  2) Public key replace Query: $\mathcal{C}$ replaces $PK_i = (ID_i, \mathbf{u}_i, \mathbf{psk}_i, SK_S\mathbf{g}, \mathbf{m}_i, \mathbf{M}_i, \mathbf{v}_i)$ to $PK_i^{'} = (ID_i, \mathbf{u}_i, \mathbf{psk}_i, *, \mathbf{m}_i, \mathbf{M}_i, *)$.
- Challenge phase: The adversary $\mathcal{A}_I$ selects and also sends two different medical message m_0, m_1 and two users (ID_S^*, ID_U^*) to $\mathcal{C}$. Then, $\mathcal{C}$ performs the following operations to reply $\mathcal{A}_I$.
  - If $ID_U^* = ID_{S'}$, $\mathcal{C}$ has acquired one of the two items $((ID_U^*, \mathbf{u}^*, *, *, *, *, *)$ or $(ID_U^*, \mathbf{u}^*, *, SK_S\mathbf{g}, \mathbf{m}_{S'}, \mathbf{M}', \mathbf{v}'))$, which means the public key $PK_{ID_U^*}$ has been replaced and has not been replaced, respectively.
    - If $PK_{ID_U^*}$ has been replaced, $\mathcal{C}$ verifies the validation of $PK_{ID_U^*}$ as below.
      - If it passes the verification, $\mathcal{C}$ updates $PK_{ID_U^*}$ to $PK_{ID_S'} = (\mathbf{m}_{ID_S'} | \mathbf{M}_{ID_S'}^\top)$. After that, $\mathcal{C}$ chooses $\epsilon \xleftarrow{\$} \mathbb{Z}_q^{n+1}$ and $\varsigma \in \{0,1\}^m$ at random. In addition, $\mathcal{C}$ sends two items $(\mathbf{m}'_{S'}, \mathbf{M}'_{S'}, \epsilon)$ and $(\mathbf{m}'_{S'}, \mathbf{M}'_{S'}, (\varsigma^\top \mathbf{m}'_{S'} | \mathbf{M}'_{S'}\varsigma))$ to $\mathcal{A}_I$. We say that $PK_{ID_S'}$ is valid if $\mathcal{A}_I$ can distinguish $(\mathbf{m}'_{S'}, \mathbf{M}'_{S'}, \epsilon)$ and $(\mathbf{m}'_{S'}, \mathbf{M}'_{S'}, (\varsigma^\top \mathbf{m}'_{S'} | \mathbf{M}'_{S'}\varsigma))$ with overwhelming probability.
      - Otherwise, $\mathcal{C}$ aborts the game.
    - If $PK_{ID_U^*}$ has not been replaced, $\mathcal{C}$ chooses $\epsilon \xleftarrow{\$} \mathbb{Z}_q^{n+1}$ and $\varsigma \xleftarrow{\$} \{0,1\}^m$ randomly. After that, $\mathcal{C}$ sends two items $(\mathbf{m}'_{S'}, \mathbf{M}'_{S'}, \epsilon)$ and $(\mathbf{m}'_{S'}, \mathbf{M}'_{S'}, (\varsigma^\top \mathbf{m}'_{S'} | \mathbf{M}'_{S'}\varsigma))$ to $\mathcal{A}_I$. We say that $PK_{ID_S'}$ is valid if $\mathcal{A}_I$ can distinguish $(\mathbf{m}'_{S'}, \mathbf{M}'_{S'}, \epsilon)$ and $(\mathbf{m}'_{S'}, \mathbf{M}'_{S'}, (\varsigma^\top \mathbf{m}'_{S'} | \mathbf{M}'_{S'}\varsigma))$ with overwhelming probability.

  Finally, $\mathcal{C}$ returns the challenge ciphertext $c^* = (\mu_1^*, \mu_2^*, \mu^*) = (\mu', \mathbf{w}^\top \mathbf{u}_{S'}, \mathbf{M}^\top \mathbf{w} + 2\mathbf{v}_U)$ to $\mathcal{A}_I$.
  - If $ID_U^* \neq ID_{S'}$, $\mathcal{C}$ terminates this game and returns $\perp$ to $\mathcal{A}_I$.
- Guess phase: Ultimately, $\mathcal{A}_I$ outputs a guess b'. Then, $\mathcal{C}$ verifies if $b' \stackrel{?}{=} b$. If it holds, $\mathcal{C}$ will output a solution of the LWE hardness; Otherwise, $\mathcal{C}$ will output $\perp$.

We define $Adv_{\mathcal{A}_I}^{\widehat{\textbf{Game 2}}}(\lambda)$ as the advantage of $\mathcal{A}_I$ wins the $\widehat{\textbf{Game 2}}$.

As for $\mathcal{A}_I$, $\widehat{\textbf{Game 2}}$ and $\widehat{\textbf{Game 1}}$ are statistically indistinguishable according to Theorem 1. Thus, we have:

$$|Adv_{\mathcal{A}_I}^{\widehat{\textbf{Game 2}}}(\lambda) - Adv_{\mathcal{A}_I}^{\widehat{\textbf{Game 1}}}(\lambda)| \leq \textbf{negl}(\lambda). \tag{32}$$

In summary, we say

$$\begin{aligned} & Adv_{LWE} - |Adv_{\mathcal{A}_I}^{\widehat{\textbf{Game 2}}}(\lambda) - \frac{1}{2}| \\ & \leq |Adv_{\mathcal{A}_I}^{\widehat{\textbf{Game 0}}}(\lambda) - Adv_{\mathcal{A}_I}^{\widehat{\textbf{Game 1}}}(\lambda)| + \\ & |Adv_{\mathcal{A}_I}^{\widehat{\textbf{Game 1}}}(\lambda) - Adv_{\mathcal{A}_I}^{\widehat{\textbf{Game 2}}}(\lambda)| \leq \textbf{negl}(\lambda). \end{aligned} \tag{33}$$

□

Theorem 4: If there exists a Type-II adversary $\mathcal{A}_{II}$ who has the ability to break IND-CCA2 of the proposed PQ-CLSC scheme with a non-negligible advantage Adv'_{LWE} in probability-polynomial time, then there exists an algorithm $\mathcal{B}_2$ can solve the LWE hardness within $Q_{KG} + Q_{PSK} + Q_{SV}$ query time, where Q_{KG}, Q_{PSK}, Q_{SV} means $\mathcal{A}_{II}$ can perform key generation query, partial secret key query, and secret value query, respectively.

Proof Suppose there exists a challenger $\mathcal{C}$ who can perform the algorithm $\mathcal{B}_2$. We finished the security analysis below.

- Setup phase: One challenger $\mathcal{C}$ executes $(pp,(mpk,msk)) \leftarrow$ **Setup**(n,λ). Then $\mathcal{C}$ transmits pp and msk to $\mathcal{A}_{II}$.
- Query phase: In this phase, $\mathcal{A}_{II}$ can access almost exactly the same queries as in the former theorem except the following. Public key request Query: After obtaining this query of user ID_i from $\mathcal{A}_{II}$, $\mathcal{C}$ checks whether $ID_i \stackrel{?}{=} ID_{S'}$. If it holds, $\mathcal{C}$ will update $\mathbf{M}_{ID_{S'}} = \mathbf{M}^*$ and $\mathbf{m}_{ID_{S'}} = \mathbf{m}^*$; Otherwise, $\mathcal{C}$ randomly chooses $\mathbf{M}_i \stackrel{\$}{\leftarrow} \mathbb{Z}_q^{n\times m}$, $\mathbf{v}_i \stackrel{\$}{\leftarrow} D^m_{\mathbb{Z},q\alpha}$, and $SK_i\mathbf{g} \stackrel{\$}{\leftarrow} D^n_{\mathbb{Z},q\alpha}$. After that, $\mathcal{C}$ computes $\mathbf{m}_i = 2\mathbf{v}_i + \mathbf{M}_i^\top SK_i\mathbf{g} \bmod q$. Eventually, $\mathcal{C}$ inserts $(ID_i,\mathbf{m}_i,\mathbf{M}_i,\mathbf{v}_i)$ into the $\mathcal{O}^{list}_{H_1}$ and then returns $(ID_i,\mathbf{m}_i,\mathbf{M}_i)$ to the adversary $\mathcal{A}_{II}$.
- Challenge phase: $\mathcal{A}_{II}$ chooses and sends two different medical messages with same length (m_0,m_1) corresponding to the signcrypt user ID^*_S and unsigncrypt user ID^*_U to $\mathcal{C}$. Then, $\mathcal{C}$ verifies if $ID^*_U \stackrel{?}{=} ID_{S'}$.
  1) If the equation holds, $\mathcal{C}$ will terminate the challenge query and return $\perp$;
  2) Otherwise, $\mathcal{C}$ accesses the list $\mathcal{O}^{list}_{H_1}$ and then executes $\mathbf{psk}_{ID_{S'}} \leftarrow$ **Partial secret key Extract**$(ID_{S'},pp)$. $\mathcal{C}$ also calculates $\mathbf{Hash}^1_{S'} = H_1(ID_{S'})$. Eventually, $\mathcal{C}$ calculates and transmits the challenge ciphertext $\mathbf{c}^* = (\mu_1^*|\mu_2^*|\mu^*) = (\mu', \mathbf{w}^\top\mathbf{u}_{S'}, \mathbf{M}^\top\mathbf{w} + 2\mathbf{v}_U)$ to $\mathcal{A}_{II}$.
- Guess: Ultimately, $\mathcal{A}_{II}$ outputs a guess b'. Then, $\mathcal{C}$ verifies if $b' \stackrel{?}{=} b$. If it holds, $\mathcal{C}$ will output a solution of the LWE hardness; Otherwise, $\mathcal{C}$ will output $\perp$. The probability Adv'_{LWE} for this theorem is analogous to the former.

□

B. Unforgeability of PQ-CLSC

Theorem 5: If there exists a Type-I adversary $\mathcal{A}_I$ who has the ability to break EU-CMA of PQ-CLSC primitive within a non-negligible advantage Adv_{SIS} in probability-polynomial time, then there exists an algorithm $\mathcal{B}_3$ can solve the SIS hardness with probability $Adv_{SIS} = Adv_{A_I}\cdot(1-2^{-\omega(\log n)})$.

Proof Assume that there exists a challenger $\mathcal{C}$ who can perform the algorithm $\mathcal{B}_3$ and an adversary $\mathcal{A}_I$ can counterfeit a ciphertext. We finished the security analysis below.

- Setup phase: A challenger $\mathcal{C}$ performs $(pp,(mpk,msk)) \leftarrow$ **Setup**(n,λ). Then $\mathcal{C}$ sends pp to $\mathcal{A}_I$ and keeps msk in secret. In this way, $\mathcal{A}_I$ knows nothing about the msk. Moreover, the challenger $\mathcal{C}$ maintains three lists $\mathcal{O}^{list}_{H_1}$, $\mathcal{O}^{list}_{H_2}$, and $\mathcal{O}^{list}_{PK}$ to record H_1 oracle, H_2 oracle, and public key oracle, respectively. These lists are initialized empty at the beginning.
- Query phase: The adversary $\mathcal{A}_I$ can access several queries and the challenger $\mathcal{C}$ then replies the corresponding response to $\mathcal{A}_I$. The query regulations are identical to Query 1 in Theorem 4.
- Forge phase: $\mathcal{A}_I$ forges and delivers $\mathbf{c}^* = (\mu_1^*|\mu_2^*|\mu^*)$ of the challenge signcrypt user and unsigncrypt user (ID^*_S, ID^*_U) to $\mathcal{C}$. We say that $\mathcal{C}$ succeeds when the challenge ciphertext is valid. Furthermore, $\mathcal{A}_I$ forges $\mathbf{c} = (\mu_1|\mu_2|\mu)$ of the challenge signcrypt user and unsigncrypt user (ID^*_S, ID^*_U). Accordingly, we have:

$$\begin{aligned}&\begin{bmatrix}\mathbf{A} & \mathbf{A} & 0\\ 0 & \mathbf{M}_U & \mathbf{M}_U\end{bmatrix}\mathbf{t}^* - \begin{bmatrix}H_1(ID^*_S, ID^*_U)\\ PK_{U^*}\end{bmatrix}\mathbf{g}^*\\ &= \begin{bmatrix}\mathbf{A} & \mathbf{A} & 0\\ 0 & \mathbf{M}_U & \mathbf{M}_U\end{bmatrix}\mathbf{t}' - \begin{bmatrix}H_1(ID^*_S, ID^*_U)\\ PK_{U^*}\end{bmatrix}\mathbf{g}'\end{aligned} \tag{34}$$

We obtain that $\mathbf{M}(\mathbf{t}^* - \mathbf{t}' + \mathbf{psk}_U(\mathbf{g}' - \mathbf{g}^*)) = 0$, where $\mathbf{t}^* \le 2\sigma\sqrt{3l}$, $\mathbf{t}' \le 2\sigma\sqrt{3l}$, $\mathbf{g}' \le \lambda$, and $\mathbf{g}^* \le \lambda$. Consequently, we can say that

$$\frac{\mathbf{t}^* - \mathbf{t}' + \mathbf{psk}_U(\mathbf{g}' - \mathbf{g}^*)}{4} \le s\lambda\sqrt{2l} + 2\sigma\sqrt{2l}$$

is satisfied with overwhelming probability.

Therefore, the probability to solve the SIS hardness is $Adv_{SIS} = Adv_{A_I}\cdot(1-2^{-\omega(\log n)})$ since the probability of $\mathbf{t}^* - \mathbf{t}' + \mathbf{psk}_U(\mathbf{g}' - \mathbf{g}^*) = 0$ is less than $(1-2^{-\omega(\log n)})$ due to the nature of lattice sampling algorithm [40].

□

Theorem 6: If there exists a Type-II adversary $\mathcal{A}_{II}$ who has the ability to break EU-CMA of the PQ-CLSC primitive within a non-negligible advantage Adv'_{SIS} in probability-polynomial time, then there exists an algorithm $\mathcal{B}_4$ can solve the SIS hardness with probability $Adv'_{SIS} = Adv_{A_{II}}\cdot(1-2^{-\omega(\log n)})$.

Proof Suppose there exists a challenger $\mathcal{C}$ who can perform the algorithm $\mathcal{B}_4$ and an adversary $\mathcal{A}_{II}$ can counterfeit a ciphertext. We finished the security analysis below.

- Setup phase: A challenger $\mathcal{C}$ performs $(pp,(mpk,msk)) \leftarrow$ **Setup**(n,λ). Then $\mathcal{C}$ sends pp to $\mathcal{A}_{II}$ and keeps msk in secret. In this way, $\mathcal{A}_I$ knows nothing about the msk. Besides, the challenger $\mathcal{C}$ maintains three lists $\mathcal{O}^{list}_{H_1}$, $\mathcal{O}^{list}_{H_2}$, and $\mathcal{O}^{list}_{PK}$, which are identical to the former theorem.
- Query phase: The adversary $\mathcal{A}_{II}$ can access several queries and the challenger $\mathcal{C}$ then replies the corresponding response to $\mathcal{A}_{II}$. The query regulations are the same as the Query phase in Theorem 5.
- Forge phase: $\mathcal{A}_{II}$ forges and delivers $\mathbf{c}^* = (\mu_1^*|\mu_2^*|\mu^*)$ of the challenge signcrypt user and unsigncrypt user (ID^*_S, ID^*_U) to $\mathcal{C}$. We say that $\mathcal{C}$ succeeds when the challenge ciphertext is not $\perp$. Moreover, $\mathcal{A}_{II}$ can also forges $\mathbf{c} = (\mu_1|\mu_2|\mu)$ of the challenge signcrypt user and unsigncrypt user (ID^*_S, ID^*_U) [45]. Thus, we have the following equalities:

$$\begin{aligned}&\begin{bmatrix}\mathbf{A} & \mathbf{A} & 0\\ 0 & \mathbf{M}_U & \mathbf{M}_U\end{bmatrix}\mathbf{t}^* - \begin{bmatrix}H_1(ID^*_S, ID^*_U)\\ PK_{U^*}\end{bmatrix}\mathbf{g}^*\\ &= \begin{bmatrix}\mathbf{A} & \mathbf{A} & 0\\ 0 & \mathbf{M}_U & \mathbf{M}_U\end{bmatrix}\mathbf{t}' - \begin{bmatrix}H_1(ID^*_S, ID^*_U)\\ PK_{U^*}\end{bmatrix}\mathbf{g}'\end{aligned} \tag{35}$$

We acquire that $\mathbf{M}(\mathbf{t}^* - \mathbf{t}' + \mathbf{psk}_U(\mathbf{g}' - \mathbf{g}^*)) = 0$, where $\mathbf{t}^* \le 2\sigma\sqrt{3l}$, $\mathbf{t}' \le 2\sigma\sqrt{3l}$, $\mathbf{g}' \le \lambda$, and $\mathbf{g}^* \le \lambda$. Hence,

$$\frac{\mathbf{t}^* - \mathbf{t}' + \mathbf{psk}_U(\mathbf{g}' - \mathbf{g}^*)}{4} \le s\lambda\sqrt{2l} + 2\sigma\sqrt{2l}$$

is satisfied with overwhelming probability.

To conclude, the probability of solving the SIS hardness is $Adv'_{SIS} = Adv_{A_I}\cdot(1-2^{-\omega(\log n)})$ since the probability of $\mathbf{t}^* - \mathbf{t}' + \mathbf{psk}_U(\mathbf{g}' - \mathbf{g}^*) = 0$ is lower than $(1-2^{-\omega(\log n)})$ due to the nature of lattice sampling algorithm [40].

□

TABLE II
SYMBOLS AND DESCRIPTION OF PERFORMANCE EVALUATION

Symbols	Description
T_{vmul}	The time of vector multiplication operation.
T_{vadd}	The time of vector additive operation.
T_{smul}	The time of scalar multiplication on bilinear pairing group.
T_{pair}	The time of pairing operation.
T_{pis}	The time of pre-image sampling algorithm.
T_{minv}	The time of modular inversion operation.
T_{htp}	The time of hash-to-point operation.
$\|G_{pair}\|$	The length of elements in bilinear pairing group.
$\|Z_q^*\|$	The length of elements in $\|Z_q^*\|$.
$\|m\|$	The size of messages.
$\|n\|$	The security parameter.
$\|q\|$	The large prime.
$\|k\|$	The integer.
$\|l\|$	The number large to $5n \log q$.

TABLE III
RUNNING TIMES OF OPERATIONS

Operation	Execution Time
T_{vmul}	5.183 (ms)
T_{vadd}	0.067 (ms)
T_{smul}	1.541 (ms)
T_{pair}	4.156 (ms)
T_{pis}	33.281 (ms)
T_{minv}	0.003 (ms)
T_{htp}	3.739 (ms)

VII. PERFORMANCE EVALUATION AND COMPARISON WITH PRIOR ARTS

In this sector, we perform a comparative analysis of our scheme with other existing signcryption schemes [18]–[24] regarding both computational overhead and communication overhead[1]. Specifically, all simulation experiments through the MATLAB operating platform were conducted in a Win 10 system environment with a processor of AMD Ryzen 7 5800H with Radeon Graphics at 3.20 GHz and running memory of 16.0 GB. In general, in Table II, all the symbols used in our efficiency analysis are given along with their specific meanings. The running time of the seven operations involved in our compared schemes is shown in Table III.

A. Communication Overhead Comparison

For the communication overhead, we focus on comparing the length of ciphertext. Table IV shows the theoretical value of ciphertext length computation in our scheme and five other existing schemes [20]–[24]. It is easy to notice that the ciphertext length in our scheme is $2kn \log^2 q$, which is significantly lower than the other lattice-based signcryption schemes [20]–[24].

B. Computational Overhead Comparison

For the comparative analysis of computational overhead, we present the theoretical computational values of signcryption and unsigncryption overhead for our primitive and the other five existing mechanisms [20]–[24] in Table V. By combining this analysis with the information in Table III, we can determine that the pre-image sample algorithm has the highest time overhead. However, in our scheme, we have successfully avoided it to minimize the time overhead. Specifically, the signcryption overhead in our scheme consists of four vector multiplication operations and five vector additive operations, while the unsigncryption overhead includes two vector multiplication operations and three vector additive operations. By referring to both Table III and Table V, we can conclude that the time overhead of our protocol is significantly lower than that of all the lattice-based schemes [20]–[24].

[1]We omit the overhead comparison of schemes [18] and [19] since these schemes are not resistant to quantum attacks (shown in Table VI).

TABLE IV
COMPARISON OF COMMUNICATION OVERHEAD

Schemes	Ciphertext Length
Li et al. [20]	$n + 6n \log^2 q$
Zhang et al. [21]	$796 + 36n^2 \log^3 q$
Yan et al. [22]	$kmn^2 \log^2 q$
Sun et al. [23]	$2mk^2 n \log^2 q$
Yang et al. [24]	$2m^2 n \log^2 q$
Our Scheme	$2kn \log^2 q$

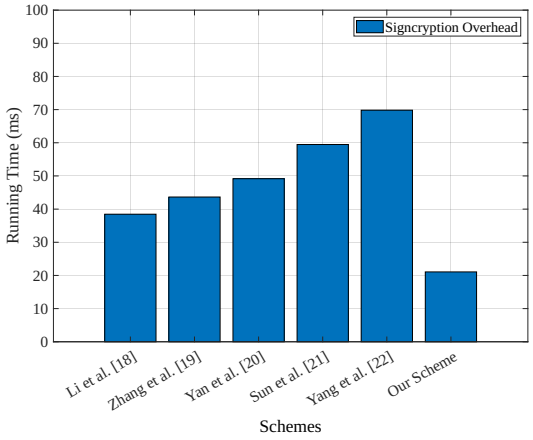

Fig. 2. Approximate Running Time Comparison of Signcryption.

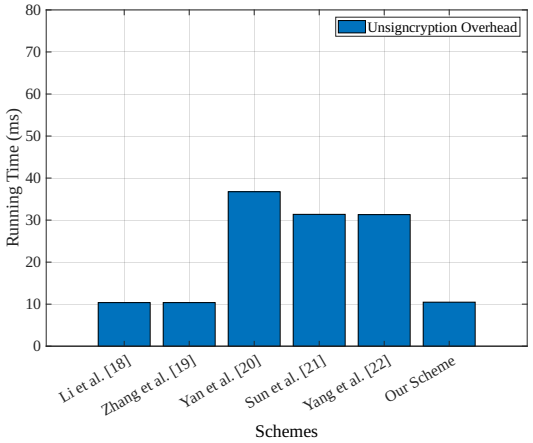

Fig. 3. Approximate Running Time Comparison of Unsigncryption.

TABLE V
COMPARISON OF COMPUTATIONAL OVERHEAD

Schemes	Signcryption Overhead	Unsigncryption Overhead
Li et al. [20]	$T_{pis} + T_{vmul}$	$2T_{vmul}$
Zhang et al. [21]	$T_{pis} + 2T_{vmul}$	$2T_{vmul}$
Yan et al. [22]	$T_{pis} + 5T_{vadd} + 3T_{vmul}$	$7T_{vadd} + 7T_{vmul}$
Sun et al. [23]	$T_{pis} + 4T_{vadd} + 5T_{vmul}$	$4T_{vadd} + 6T_{vmul}$
Yang et al. [24]	$T_{pis} + 4T_{vadd} + 7T_{vmul}$	$3T_{vadd} + 6T_{vmul}$
Our Scheme	$5T_{vadd} + 4T_{vmul}$	$3T_{vadd} + 2T_{vmul}$

TABLE VI
COMPARISON OF SECURITY PROPERTIES

Schemes	IND-CCA2	UF-CMA	Quantum Resistance	Practicality
Yu et al. [18]	✗	✗	✗	✓
Chen et al. [19]	✓	✓	✗	✓
Li et al. [20]	✓	✓	✓	✗
Zhang et al. [21]	✓	✓	✓	✗
Yan et al. [22]	✗	✗	✓	✗
Sun et al. [23]	✓	✓	✓	✗
Yang et al. [24]	✓	✓	✓	✗
Our Scheme	✓	✓	✓	✓

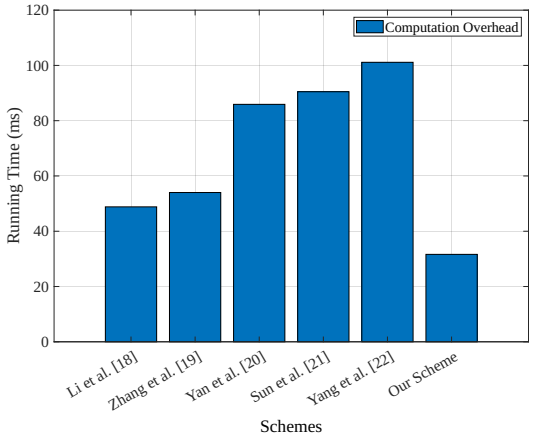

Fig. 4. Approximate Running Time Comparison of Computation Overhead.

Through the MALTAB experimental platform, we conducted simulation experiments for our scheme and the other five signcryption protocols [20]–[24] to further comprehensively demonstrate the comparison analysis findings in terms of computational overhead. The signcryption overheads of our scheme and other schemes [20]–[24] are shown in Fig. 2. Combining Table III and Table V, we can calculate that the signcryption overhead of our scheme is $5\times T_{vadd}+4\times T_{vmul} = 5\times 0.067+4\times 5.183 = 21.067(ms)$. From Fig. 2, we observe that the signcryption overhead of our scheme is considerably lower than five existing lattice-based signcryption schemes [20]–[24]. In particular, our signcryption overhead is between 0.30 to 0.55 times of them [20]–[24].

The comparison of unsigncryption overhead is depicted in Fig. 3. In particular, the unsigncryption overhead of our scheme is essentially the same as schemes [20] and [21]. While the unsigncryption overheads of schemes [22], [23] and [24] are obviously higher than schemes [20], [21] and ours. Especially, the calculation of the unsigncryption cost can also show similar results to the above simulation experiments. The overheads of schemes [20], [21] and our scheme are calculated as $2 \times T_{vmul} = 2 \times 5.183 = 10.366(ms)$,$2 \times T_{vmul} = 2 \times 5.183 = 10.366(ms)$, and $3 \times T_{vadd} + 2 \times T_{vmul} = 3 \times 0.067 + 2 \times 5.183 = 10.567(ms)$, respectively. Compared with other lattice-based schemes [20]–[24], the overhead of our scheme is 0.28 to 1 times that of existing schemes.

A comparative analysis of the overall computational overhead is shown in Fig. 4. Concretely, for our scheme, the computational overhead is $5\times T_{vadd}+4\times T_{vmul}+3\times T_{vadd}+2 \times T_{vmul} = 5 \times 0.067 + 4 \times 5.183 + 3 \times 0.067 + 2 \times 5.183 = 31.634(ms)$. Consequently, incorporating theoretical value calculations and simulation experiments, it is clear that the computational overhead of our scheme is marginally lower than other lattice-based signcrytion schemes [22]–[24] and noticeably lower than schemes [20], [21]. In summary, the computational overhead of our scheme is dramatically lower than the other five lattice-based signcryption schemes, being 0.31 to 0.64 times that of their schemes [20]–[24].

C. Security Comparison

As far as the security of the scheme is concerned, we mainly consider the four components IND-CCA2, UF-CMA, quantum resistance, and practicality. Seen from Table VI, we find that only scheme [18] and scheme [22] fail to meet the security requirements of IND-CCA2 and UF-CMA. For the property of Quantum Resistance, all comparison schemes, except scheme [18] and scheme [19], are capable of resisting quantum attacks. Except for our scheme, all schemes with quantum resistance security are impractical. In summary, our scheme guarantees practicality while fulfilling the fullest security requirements.

VIII. Conclusion

In this paper, we propose a new post-quantum certificateless signcryption primitive, called PQ-CLSC. It enables medical data transmission safely in the IoMT while resistant to the quantum computing attacks. We begin by presenting the system models and security models. After that, we illustrate our designed mechanism in detail. The proposed PQ-CLSC undergoes rigorous security analysis, demonstrating its satisfaction with IND-CCA2 and EU-CMA security in a quantum setting. We also conduct extensive experimental evaluations and comparisons, which reveal the efficiency of our protocol. These results highlight the superior practicality of our scheme compared to most state-of-the-art protocols.

References

[1] W. Mao, P. Jiang, and L. Zhu, "Locally verifiable batch authentication in iomt," *IEEE Transactions on Information Forensics and Security*, 2023.

[2] X. Chen, D. He, M. K. Khan, M. Luo, and C. Peng, "A secure certificateless signcryption scheme without pairing for internet of medical things," *IEEE Internet of Things Journal*, vol. 10, no. 10, pp. 9136–9147, 2022.

[3] X. Chen, S. Xu, T. Qin, Y. Cui, S. Gao, and W. Kong, "Aq–abs: Anti-quantum attribute-based signature for emrs sharing with blockchain," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2022, pp. 1176–1181.

[4] M. Wazid, A. K. Das, S. Shetty, J. J. Rodrigues, and M. Guizani, "Aiscm-fh: Ai-enabled secure communication mechanism in fog computing-based healthcare," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 319–334, 2022.

[5] H. Li, K. Yu, B. Liu, C. Feng, Z. Qin, and G. Srivastava, "An efficient ciphertext-policy weighted attribute-based encryption for the internet of health things," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 5, pp. 1949–1960, 2021.

[6] Y. Bao, W. Qiu, P. Tang, and X. Cheng, "Efficient, revocable, and privacy-preserving fine-grained data sharing with keyword search for the cloud-assisted medical iot system," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 5, pp. 2041–2051, 2021.

[7] M. Papaioannou, M. Karageorgou, G. Mantas, V. Sucasas, I. Essop, J. Rodriguez, and D. Lymberopoulos, "A survey on security threats and countermeasures in internet of medical things (iomt)," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 6, p. e4049, 2022.

[8] X. Chen, S. Xu, Y. Cao, Y. He, and K. Xiao, "Aqrs: Anti-quantum ring signature scheme for secure epidemic control with blockchain," *Computer Networks*, vol. 224, p. 109595, 2023.

[9] X. Chen, S. Xu, Y. He, Y. Cui, J. He, and S. Gao, "Lfs-as: lightweight forward secure aggregate signature for e-health scenarios," in *ICC 2022-IEEE International Conference on Communications*. IEEE, 2022, pp. 1239–1244.

[10] G. Xu, S. Xu, Y. Cao, F. Yun, Y. Cui, Y. Yu, K. Xiao *et al.*, "Ppseb: a postquantum public-key searchable encryption scheme on blockchain for e-healthcare scenarios," *Security and Communication Networks*, vol. 2022, 2022.

[11] Y. Zheng, "Digital signcryption or how to achieve cost (signature & encryption) cost (signature)+ cost (encryption)," in *Advances in Cryptology—CRYPTO'97: 17th Annual International Cryptology Conference Santa Barbara, California, USA August 17–21, 1997 Proceedings 17*. Springer, 1997, pp. 165–179.

[12] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *International conference on the theory and application of cryptology and information security*. Springer, 2003, pp. 452–473.

[13] M. Barbosa and P. Farshim, "Certificateless signcryption," in *Proceedings of the 2008 ACM symposium on Information, computer and communications security*, 2008, pp. 369–372.

[14] Y. Zhou, R. Xu, Z. Qiao, B. Yang, Z. Xia, and M. Zhang, "An anonymous and efficient multi-message and multi-receiver certificateless signcryption scheme for vanet," *IEEE Internet of Things Journal*, 2023.

[15] I. Ali, Y. Chen, J. Li, A. Wakeel, C. Pan, and N. Ullah, "Efficient offline/online heterogeneous-aggregated signcryption protocol for edge computing-based internet of vehicles," *IEEE Transactions on Intelligent Transportation Systems*, 2023.

[16] Y. Hou, Y. Cao, H. Xiong, Y. Song, and L. Xu, "An efficient online/offline heterogeneous signcryption scheme with equality test for iovs," *IEEE Transactions on Vehicular Technology*, 2023.

[17] Y. Yang, L. Zhang, Y. Zhao, K.-K. R. Choo, and Y. Zhang, "Privacy-preserving aggregation-authentication scheme for safety warning system in fog-cloud based vanet," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 317–331, 2022.

[18] H. Yu and B. Yang, "Pairing-free and secure certificateless signcryption scheme," *The Computer Journal*, vol. 60, no. 8, pp. 1187–1196, 2017.

[19] J. Chen, L. Wang, M. Wen, K. Zhang, and K. Chen, "Efficient certificateless online/offline signcryption scheme for edge iot devices," *IEEE Internet of Things Journal*, vol. 9, no. 11, pp. 8967–8979, 2021.

[20] F. Li, F. T. Bin Muhaya, M. K. Khan, and T. Takagi, "Lattice-based signcryption," *Concurrency and Computation: Practice and Experience*, vol. 25, no. 14, pp. 2112–2122, 2013.

[21] X. Zhang, C. Xu, and J. Xue, "Efficient multi-receiver identity-based signcryption from lattice assumption," *International Journal of Electronic Security and Digital Forensics*, vol. 10, no. 1, pp. 20–38, 2018.

[22] J. Yan, L. Wang, M. Li, H. Ahmad, J. Yue, and W. Yao, "Attribute-based signcryption from lattices in the standard model," *IEEE Access*, vol. 7, pp. 56 039–56 050, 2019.

[23] Y. Sun and W. Zheng, "An identity-based ring signcryption scheme in ideal lattice." *J. Netw. Intell.*, vol. 3, no. 3, pp. 152–161, 2018.

[24] X. Yang, H. Cao, W. Li, and H. Xuan, "Improved lattice-based signcryption in the standard model," *IEEE Access*, vol. 7, pp. 155 552–155 562, 2019.

[25] S. Sato and J. Shikata, "Lattice-based signcryption without random oracles," in *Post-Quantum Cryptography: 9th International Conference, PQCrypto 2018, Fort Lauderdale, FL, USA, April 9-11, 2018, Proceedings 9*. Springer, 2018, pp. 331–351.

[26] H. Q. Le, D. H. Duong, P. S. Roy, W. Susilo, K. Fukushima, and S. Kiyomoto, "Lattice-based signcryption with equality test in standard model," *Computer Standards & Interfaces*, vol. 76, p. 103515, 2021.

[27] J. Malone-Lee, "Identity-based signcryption," *Cryptology ePrint Archive*, 2002.

[28] L. Chen and J. Malone-Lee, "Improved identity-based signcryption," in *International workshop on public key cryptography*. Springer, 2005, pp. 362–379.

[29] Z. Liu, Y. Hu, X. Zhang, and H. Ma, "Certificateless signcryption scheme in the standard model," *Information Sciences*, vol. 180, no. 3, pp. 452–464, 2010.

[30] S. Miao, F. Zhang, S. Li, and Y. Mu, "On security of a certificateless signcryption scheme," *Information Sciences*, vol. 232, pp. 475–481, 2013.

[31] J. Yan, L. Wang, L. Wang, Y. Yang, W. Yao *et al.*, "Efficient lattice-based signcryption in standard model," *Mathematical Problems in Engineering*, vol. 2013, 2013.

[32] H. YU and N. WANG, "Certificateless proxy signcryption scheme from lattice," *Journal of Electronics Information Technology*, vol. 44, no. 7, pp. 2584–2591, 2022.

[33] H. Yu and J. Shi, "Certificateless multi-source signcryption with lattice," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 10, pp. 10 157–10 166, 2022.

[34] H. Yu, W. Wang, and Q. Zhang, "Certificateless anti-quantum ring signcryption for network coding," *Knowledge-Based Systems*, vol. 235, p. 107655, 2022.

[35] S. Vishnu, S. J. Ramson, and R. Jegan, "Internet of medical things (iomt)-an overview," in *2020 5th international conference on devices, circuits and systems (ICDCS)*. IEEE, 2020, pp. 101–104.

[36] A. Ghubaish, T. Salman, M. Zolanvari, D. Unal, A. Al-Ali, and R. Jain, "Recent advances in the internet-of-medical-things (iomt) systems security," *IEEE Internet of Things Journal*, vol. 8, no. 11, pp. 8707–8718, 2020.

[37] B. Zhang, "A lightweight data aggregation protocol with privacy-preserving for healthcare wireless sensor networks," *IEEE Systems Journal*, vol. 15, no. 2, pp. 1705–1716, 2020.

[38] M. Ajtai, "Generating hard instances of lattice problems," in *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, 1996, pp. 99–108.

[39] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," *Journal of the ACM (JACM)*, vol. 56, no. 6, pp. 1–40, 2009.

[40] D. Micciancio and C. Peikert, "Trapdoors for lattices: Simpler, tighter, faster, smaller." in *Eurocrypt*, vol. 7237. Springer, 2012, pp. 700–718.

[41] C. Gentry, C. Peikert, and V. Vaikuntanathan, "Trapdoors for hard lattices and new cryptographic constructions," in *Proceedings of the*

fortieth annual ACM symposium on Theory of computing, 2008, pp. 197–206.

[42] M. Obremski and M. Skórski, "Inverted leftover hash lemma," in *2018 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2018, pp. 1834–1838.

[43] Q. Huang and D. S. Wong, "Generic certificateless encryption in the standard model," in *Advances in Information and Computer Security: Second International Workshop on Security, IWSEC 2007, Nara, Japan, October 29-31, 2007. Proceedings 2*. Springer, 2007, pp. 278–291.

[44] H. Yu, L. Bai, M. Hao, and N. Wang, "Certificateless signcryption scheme from lattice," *IEEE Systems Journal*, vol. 15, no. 2, pp. 2687–2695, 2020.

[45] M. Jiang, Y. Hu, H. Lei, B. Wang, and Q. Lai, "Lattice-based certificateless encryption scheme," *Frontiers of Computer Science*, vol. 8, pp. 828–836, 2014.