symmetry

MDPI

Article

Quantum-Resistant Identity-Based Signature with Message Recovery and Proxy Delegation

Xiuhua Lu [1,2], Qiaoyan Wen [1], Wei Yin [1], Kaitai Liang [3], Zhengping Jin [1], Emmanouil Panaousis [4] and Jiageng Chen [5,6,*]

1 State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China; luxiuhua2015@163.com (X.L.); wqy@bupt.edu.cn (Q.W.); yinwei24005@bupt.edu.cn (W.Y.); zhpjin@bupt.edu.cn (Z.J.)
2 Faculty of Mathematics and Information Science, Langfang Normal University, Langfang 065000, China
3 Department of Computer Science, University of Surrey, Guildford GU2 7XH, UK; k.liang@surrey.ac.uk
4 Surrey Centre of Cyber Security, University of Surrey, Guildford GU2 7XH, UK; e.panaousis@surrey.ac.uk
5 School of Computer Science, Central China Normal University, Wuhan 430079, China
6 Central China Normal University Wollongong Joint Institute, Central China Normal University, Wuhan 430079, China
* Correspondence: chinkako@gmail.com

Received: 12 January 2019; Accepted: 17 February 2019; Published: 20 February 2019

Abstract: Digital signature with proxy delegation, which is a secure ownership enforcement tool, allows an original signer to delegate signature rights to a third party called proxy, so that the proxy can sign messages on behalf of the original signer. Many real-world applications make use of this secure mechanism, e.g., digital property transfer. A traditional digital signature mechanism is required to bind a message and its signature together for verification. This may yield extra cost in bandwidth while the sizes of message and signature are relatively huge. Message recovery signature, enabling to reduce the cost of bandwidth, embeds a message into the corresponding signature; therefore, only the signature will be transmitted to the verifier and the message can further be recovered from the signature. In this paper, we, for the first time, propose a novel digital signature scheme in the identity-based context with proxy delegation and message recovery features and, more importantly, our scheme is quantum resistant, in a particular lattice-based signature. Our scheme achieves delegation information and signature existential unforgeability against adaptive chosen warrant and identity. Compared with the seminal lattice-based message recovery signature, our scheme is independent from public key infrastructure, realizes delegation transfer of signature rights, and compresses signature length ulteriorly. To the best of our knowledge, this paper is the first of its type.

Keywords: quantum resistant; lattice-based; proxy delegation; message recovery; small integer solution problem; learning with errors; compression

1. Introduction

Digital signature aims at message authenticity, which can be verified by everyone with a message/signature pair. Considering the practical application, a digital signature also needs to have special properties for special functionality requirements, such as signature with delegation functionality—proxy signature. Proxy signature, which was first proposed by Mambo [1], allows an original signer to delegate his signing right to a proxy signer, so that the proxy signer can sign a message on behalf of the original signer. Proxy signature is suitable for the case where the original signer is temporarily absent so that the proxy is delegated to make a signature on behalf of the original signer. It has many real-world applications (e.g., digital property transfer) and practical variants in the literature

(e.g., [2]). We note that there have been some research works by far related to proxy delegation, such as [3–9], in which they focus on decryption delegation. This paper deals with signature delegation.

Message recovery signature is a kind of digital signature with message recovery property, and was first proposed by Nyberg and Rueppel in [10]. Compared with the traditional digital signature, a message can be embedded into the signature. As a result, only the signature itself is required in the verification stage instead of the message and signature pair in the traditional version. It reduces the amount of information to be transmitted, and thus can save the transmission bandwidth dramatically.

Combining a message recovery signature and a proxy signature, a proxy signature with message recovery emerges, which owns a hidden message and the functionality of signing right delegation transfer. Furthermore, in order to simplify key management, Singh [11] combined identity-based signature with proxy signature with message recovery, and introduced the concept of identity-based proxy signature scheme with message recovery. Such scheme can work without the existence of public key infrastructure, and the legitimacy of the user's public key is not required to be verified.

1.1. Related Work

Many researchers have paid attention to proxy signature with message recovery, and a lot of contributions [12–16] have been proposed in the literature. The schemes introduced in [12,13,15,16] are based on a discrete logarithm problem, the one proposed in [14] is based on a decisional Diffie–Hellman problem and a computational Diffie-Hellman problem. However, all these problems are solvable with a quantum computer [17], so that security of schemes [12–16] will be unreliable in the quantum era, and it is significant to construct a quantum-resistible proxy signature with message recovery.

Lattice-based cryptography is an excellent branch of post-quantum cryptography. For almost two decades, lattice-based cryptography has been on the fast track of development. Some unsolved questions in traditional cryptography, such as construction of a fully homomorphic encryption scheme [18], have found their realization in lattice-based cryptography. Due to a reliable security guarantee and powerful functionality, lattice-based cryptography becomes the preferred tool for our topic—an identity-based proxy signature scheme with message recovery.

Lattice signature is the building foundation of our topic. In 2008, Gentry et al. [19] designed the first provably secure lattice signature scheme. In 2012, Micciancio et al. [20] proposed a new trapdoor generation algorithm and gave a lattice signature scheme with better efficiency and security. In the same year, Lyubashevsky [21] gave a lattice signature scheme with better efficiency following a special lattice with simpler computations. In 2014, Bai et al. [22] proposed an improved compression technique for lattice signature in [21]. Lattice signatures in [19,20] and [21,22] are two main frames for lattice signature schemes, and the latter is with better performance.

Lattice signature schemes [19–22] are all basic signature schemes. We will consider message recovery and delegation of signing right in identity-based environment. In 2013, Tian et al. [23] proposed lattice-based message recovery signature scheme following [21]. His scheme is based on public key infrastructure without expressing delegation of signing rights. In 2016, Wang Li [24] proposed an identity-based proxy signature scheme in lattice, which follows the idea of [19] and doesn't hide messages. In 2017, Faguo Wu et al. [25] gave a lattice proxy signature with message recovery based on public key infrastructure.

1.2. Our Contribution

In this paper, we build an efficient and secure identity-based proxy signature scheme with message recovery in lattice-based cryptography. Our scheme is based on the lattice signature without trapdoors [21]. Inspired by the signature compression technique in [22], we introduce the random error matrix E_{id} with enough small entries, let $(A|I)\begin{pmatrix} S_{id} \\ E_{id} \end{pmatrix} = AS_{id} + E_{id} = H_1(id)$. According to the learning with errors problem, we keep S_{id} instead of (S_{id}, E_{id}), as the secret key of user id. Correspondingly, the signature is $S_{id}c + y$ rather than $(S_{id}c + y, E_{id}c + y)$ in our scheme. These

operations add more randomness to user secret key extraction, and reduce signature length with $E_{id}c + y$.

For proxy signature, we change the traditional idea that the original signer generates the delegated secret key and passes it to the proxy signer through the secure channel. Following the idea of two-party signature in [26], our delegated secret key is obtained with the help of proxy signer's secret key and original signer's public delegation information. Therefore, delegated secret key extraction is controlled by the proxy signer and original signer, and no secure channel is required between them. Moreover, anyone can verify the validity of delegation information because it is public.

Speaking of message recovery, we adopt the technique in [23]. Compared with the scheme in [23], our scheme takes the following three advantages. Firstly, our scheme is identity-based and does not rely on public key infrastructure maintenance. Secondly, our scheme realizes delegation transfer of signing rights. Thirdly, our scheme condenses signature length. The comparing details of two schemes are described in Section 5.

In addition, we divide the security definition in [11] into two factors: delegation information existential unforgeability against adaptive chosen warrant and identity, signature existential unforgeability against adaptive chosen message and identity. The former guarantees delegation information is credible, and the latter guarantees that proxy signature is credible. Our security definition is more comprehensive.

The rest of the paper is organized as follows. We present an overview of background knowledge in Section 2. Then, we propose our model and security definitions for an identity-based proxy signature scheme with message recovery in Section 3. In Section 4, we provide the identity-based proxy signature scheme with message recovery in lattice-based cryptography. Correctness, security, and performance analysis are discussed in Section 5. Finally, we conclude this paper in Section 6.

2. Preliminaries

2.1. Notations

$\mathbb{Z}$ is the set of integers, and $\mathbb{N}$ is the set of natural numbers. Let q be a polynomial-size prime number, $\mathbb{Z}_q$ is the set of integers in $(-q/2, q/2]$. For $a \in \mathbb{Z}$ and $d \in \mathbb{N}$, $[a]_{2^d} \in (-2^{d-1}, 2^{d-1}]$ is the unique integer satisfying $a \equiv [a]_{2^d} \left(\mathrm{mod} 2^d\right)$, $\lfloor a_d \rceil = (a - [a]_{2^d})/2^d$. For $e \in \mathbb{Z}^m$, $e_{(i)}$ is the i-th entry of e, $\|e\| = \|e\|_2 = \sqrt{\sum_{i=1}^{m} e_{(i)}^2}$ is the Euclidean norm of e, and $\|e\|_\infty = \max_{1 \le i \le m} \left|e_{(i)}\right|$. For matrix $T \in \mathbb{Z}^{m \times n}$, $T(i,j)$ is the entry in i-th row and j-th column, $\|T\|$ is the largest Euclidean norm of its column vectors, and $\tilde{T}$ is its Gram–Schmidt orthogonalization. If s_1 and s_2 are two bit strings, $s_1||s_2$ is their concatenation, $s_1 \oplus s_2$ is the result of xor computation. In addition, $|s_1|^{l_1}$ is the prefix of s_1 with length l_1, $|s_1|_{l_2}$ is the suffix of s_1 with length l_2.

2.2. Lattice Theory

In this subsection, basic concepts and major algorithms related to our scheme are illustrated. For readers who are interested in details, please see literature [19,27,28].

Definition 1. *Algorithm TrapGen(q, m), with $m \ge 5n \log q$, outputs a pair (A, T) which satisfies the following conditions: 1. $A \in \mathbb{Z}_q^{n \times m}$ follows uniform distribution with overwhelming probability; 2. $T \in \mathbb{Z}^{m \times m}$, $||T|| \le O(n \log q)$ and $||\tilde{T}|| \le O(\sqrt{n \log q})$ 3. $AT = 0(mod q)$*

Definition 2. *$\mathbb{D}_\sigma$ is a discrete Gaussian distribution on $\mathbb{Z}$, with center 0 and standard deviation σ. $\mathbb{D}_\sigma^{m \times n}$ is a matrix with m rows and n columns, and every entry in the matrix follows the distribution $\mathbb{D}_\sigma$.*

Definition 3. *For $A \in \mathbb{Z}_q^{n\times m}$, a short basis T of $\Lambda_q^{\perp}(A)$, $u \in \mathbb{Z}_q^n$, and Gaussian parameter $\sigma \geq \left\|\tilde{T}\right\| \cdot \omega\left(\sqrt{\log m}\right)$, algorithm SamplePre$(A, T, u, \sigma)$ outputs some $e \in \mathbb{Z}^m$ such that $||e|| \leq \sigma\sqrt{m}$ and $Ae = u \,(\mathrm{mod}\, q)$.*

Definition 4. *Given a uniform random matrix $A \in \mathbb{Z}_q^{n\times m}$, the small integer solution (SIS) problem is to find a short vector $v \in \mathbb{Z}^m$, such that $Av = 0 \,(\mathrm{mod}\, q)$ and $||v|| \leq \beta$ for some appropriate parameter β.*

Definition 5. *Given a pair $\left(A, A^{\top}s + e\right)$, where $A \in \mathbb{Z}_q^{n\times m}$ follows uniform distribution with overwhelming probability, $s \leftarrow \mathbb{D}_{\sigma}^n$, $e \leftarrow \mathbb{D}_{\sigma}^m$ for appropriate parameter σ, the learning with errors (LWE) problem is to find s.*

With appropriate parameters, LWE and SIS problems are notably hard average problems in lattice theory, and they are the security basis of most cryptographic systems in lattice.

3. Identity-Based Proxy Signature with Message Recovery

Our model and security definitions for an identity-based proxy signature scheme with message recovery (IDPSWM) come from the literature [11], and two adjustments are made.

- In our model, the delegation information is public, everyone may verify its legality; whereas, in [11], the delegation information is sent to the proxy signer secretly, and only the proxy signer can verify its legality. Therefore, a secure channel is unnecessary to transmit delegation information in our model, and every user can verify delegation information legality.
- To make it easier to understand, we divide scheme security into two factors: delegation information existential unforgeability against adaptive chosen warrant and identity (EUF-ID-CWA), signature existential unforgeability against adaptive chosen message and identity (EUF-ID-CMA). EUF-ID-CWA security assures that delegation information is believable. EUF-ID-CMA security assures that signature is believable.

3.1. Our Model

There are three types of users: the original signer, the proxy signer, and the verifier, as well as a private key generator (PKG) in the system; their roles are as follows:

- $Setup\,(n)$: PKG inputs the security parameter n, outputs system public parameters *params* and the system secret master key *msk*.
- $KeyExtract\,(msk, id)$: Given an identity id, PKG makes use of the system secret master key *msk* and provides the secret key sk_{id} for the identity id.
- $DelGen\left(sk_{id_O}, id_P, w\right)$: The original signer id_O inputs his secret key sk_{id_O}, and the warrant w associated with proxy signer id_P, computes the delegation $W_{O\to P}$, and publishes delegation information $d_g = (id_O, id_P, w, W_{O\to P})$ to all system users.
- $DelVer\left(d_g = (id_O, id_P, w, W_{O\to P})\right)$: For arbitrary system users, he verifies the legality of delegation information $d_g = (id_O, id_P, w, W_{O\to P})$. If it is legal, the output is 1, the delegation is accepted; otherwise, the output is 0, and the delegation is rejected.
- $PkeyGen\left(sk_{id_P}, d_g = (id_O, id_P, w, W_{O\to P})\right)$: The proxy signer id_P verifies whether the delegation information $d_g = (id_O, id_P, w, W_{O\to P})$ is valid. If it is invalid, he rejects this delegation. Otherwise, he inputs his secret key sk_{id_P} and the delegation information $d_g = (id_O, id_P, w, W_{O\to P})$, outputs the delegated secret key $sk_{O,P,w}$.
- $PSign\,(sk_{O,P,w}, \varpi)$: The proxy signer id_P inputs his delegated secret key $sk_{O,P,w}$ and the message ϖ, outputs the proxy signature ς.
- $PVer\left(d_g = (id_O, id_P, w, W_{O\to P}), \varsigma\right)$: For arbitrary system users, he first recovers the message ϖ associated with signature ς, and then verifies the legality of the message/ signature pair (ϖ, ς) with regard to $d_g = (id_O, id_P, w, W_{O\to P})$. If it is legal, the output is 1, the message is accepted; otherwise, the output is 0, and the message is rejected.

As to scheme correctness, seven algorithms should satisfy the following rules: For every security parameter n, $(params, msk) \leftarrow Setup(n)$, $sk_{id} \leftarrow KeyExtract(msk, id)$, $d_g = (id_O, id_P, w, W_{O\to P}) \leftarrow DelGen\left(sk_{id_O}, id_P, w\right)$, $sk_{O,P,w} \leftarrow PkeyGen\left(sk_{id_P}, d_g = (id_O, id_P, w, W_{O\to P})\right)$, $\varsigma \leftarrow PSign\left(sk_{O,P,w}, \varpi\right)$, the probability of $1 \leftarrow PVer\left(d_g = (id_O, id_P, w, W_{O\to P}), \varsigma\right)$ is overwhelming.

3.2. Security Definitions

Scheme security includes two factors: delegation information existential unforgeability against adaptive chosen warrant and identity (EUF-ID-CWA), signature existential unforgeability against adaptive chosen message and identity (EUF-ID-CMA).

3.2.1. EUF-ID-CWA

EUF-ID-CWA security is described by the next game between a challenger $\mathcal{C}$ and a forger $\mathcal{F}$.

- Initial Phase: The challenger $\mathcal{C}$ runs $Setup$ algorithm to get system public parameters *params* and the system secret master key *msk*. $\mathcal{C}$ returns *params* to the forger $\mathcal{F}$ and keeps *msk* himself.
- Query Phase: The forger $\mathcal{F}$ makes the following queries adaptively with a polynomial bounded number, and the challenger $\mathcal{C}$ has the obligation to make reasonable answers.

  1. $KeyExtract(id)$: $\mathcal{F}$ selects a user identity id, sends it to the challenger $\mathcal{C}$. $\mathcal{C}$ invokes algorithm $KeyExtract(msk, id)$ to get the associated secret key sk_{id}. Then, $\mathcal{C}$ returns sk_{id} to $\mathcal{F}$.

  2. $DelGen(id_O, id_P, w)$: $\mathcal{F}$ selects the original signer id_O, the proxy signer id_P, and the warrant w, and sends all of them to the challenger $\mathcal{C}$. $\mathcal{C}$ executes $KeyExtract(id_O)$ query to get the associated secret key sk_{id_O}, and then invokes algorithm $DelGen(sk_{id_O}, id_p, w)$ to get $W_{O\to P}$ and returns it to $\mathcal{F}$.
- Forge Phase: The forger $\mathcal{F}$ gives his forgery $\left(d_g = (id_O, id_P, w, W_{O\to P})\right)$. If the following conditions are satisfied: $DelVer\left(d_g = (id_O, id_P, w, W_{O\to P})\right) = 1$, id_O doesn't occur in the $KeyExtract$ query, (id_O, id_P, w) doesn't occur in the $DelGen$ query, and his attack is successful.

  Let ε_1 be the success probability of $\mathcal{F}$ in this game.

Definition 6. *An identity-based proxy signature scheme with message recovery (IDPSWM) is delegation information existentially unforgeable against adaptive chosen warrant and identity (EUF-ID-CWA), if for every polynomial time forger $\mathcal{F}$, ε_1 is negligible.*

3.2.2. EUF-ID-CMA

EUF-ID-CMA security is demonstrated by the following game between a challenger $\mathcal{C}$ and a forger $\mathcal{F}$.

- Initial Phase: The challenger $\mathcal{C}$ runs the $Setup$ algorithm to get system public parameters *params* and the system secret master key *msk*. $\mathcal{C}$ returns *params* to the forger $\mathcal{F}$ and keeps *msk* secret.
- Query Phase: The forger $\mathcal{F}$ executes the following queries adaptively with a polynomial bounded number, and the challenger $\mathcal{C}$ has to return reasonable answers.

  1. $KeyExtract(id)$: $\mathcal{F}$ selects a user identity id and sends it to the challenger $\mathcal{C}$. $\mathcal{C}$ invokes algorithm $KeyExtract(msk, id)$ to get secret key sk_{id}. Then, $\mathcal{C}$ returns sk_{id} to $\mathcal{F}$.

  2.$DelGen(id_O, id_p, w)$: $\mathcal{F}$ selects the original signer id_O, the proxy signer id_P, and the warrant w, submits them to the challenger $\mathcal{C}$. $\mathcal{C}$ executes $KeyExtract(id_0)$ query to get the associated secret sk_{id_O}, and then invokes algorithm $DelGen(sk_{id_O}, id_p, w)$ to get $W_{O\to P}$ and returns it to $\mathcal{F}$.

  3. $PkeyGen(d_g = (id_O, id_P, w, W_{O\to P}))$: $\mathcal{F}$ sends the delegation information $d_g = (id_O, id_P, w, W_{O\to P})$ to the challenger $\mathcal{C}$. $\mathcal{C}$ verifies its validity firstly. If it isn't valid, he refuses to

respond. Otherwise, $\mathcal{C}$ executes a $KeyExtract\ (id_P)$ query to get secret key sk_{id_P}, invokes algorithm $PkeyGen\ \left(sk_{id_P}, d_g = (id_O, id_P, w, W_{O\to P})\right)$ to get delegated secret key $sk_{O,P,w}$ and returns it to $\mathcal{F}$.

4. $PSign\ \left(d_g = (id_O, id_P, w, W_{O\to P}), \varpi\right)$: $\mathcal{F}$ submits $d_g = (id_O, id_p, w, W_{O\to P})$ and message ϖ to the challenger $\mathcal{C}$. $\mathcal{C}$ verifies the legality of $d_g = (id_O, id_p, w, W_{O\to P})$. If it is illegal, $\mathcal{C}$ rejects answering the query. Otherwise, he executes the $PkeyGen(d_g = (id_O, id_p, w, W_{O\to P}))$ query to get the delegated secret key $sk_{O,P,w}$, invokes algorithm $PSign(sk_{O,P,w}, \varpi)$ to get signature ς, and returns it to $\mathcal{F}$.

- Forge Phase: The forger $\mathcal{F}$ gives his forgery $\left(d_g = (id_O, id_P, w, W_{O\to P}), \varsigma\right)$.

Recovering the message ϖ from ς, if the following conditions hold: $PVer\ \left(d_g = (id_O, id_P, w, W_{O\to P}), \varsigma\right) = 1$, $d_g = (id_O, id_P, w, W_{O\to P})$ doesn't occur in the $PkeyGen$ query, $\left(d_g = (id_O, id_P, w, W_{O\to P}), \varpi\right)$ doesn't occur in the $PSign$ query, his attack is successful.

Let ε_2 be the success probability of $\mathcal{F}$ in the game.

Definition 7. *An identity-based proxy signature scheme with message recovery (IDPSWM) is signature existentially unforgeable against the adaptive chosen message and identity (EUF-ID-CMA), if, for every polynomial time forger $\mathcal{F}$, ε_2 is negligible.*

4. Our Scheme

In this section, we introduce our identity-based proxy signature scheme with message recovery from lattice assumption. Our scheme includes seven algorithms, which also can be seen from Figure 1.

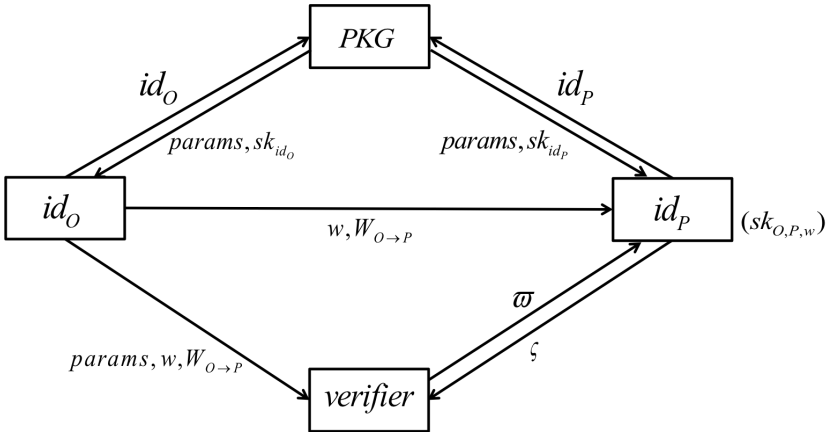

Figure 1. Flow chart of our signature algorithm.

- $Setup(n)$: Inputting the security parameter n, PKG works as follows:

1. Invoke $TrapGen\ (q, m)$ algorithm to obtain a pair of matrices $(A \in \mathbb{Z}_q^{n\times m}, T \in \mathbb{Z}^{m\times m})$.

2. Let $H_1 : \{0,1\}^* \to \mathbb{Z}_q^{n\times n}$ be a secure hash function.

3. Let $H_2, H_5 : \{0,1\}^* \to \{-1,0,1\}^n$ be secure hash functions, and the image Hamming weight is not larger than λ_1.

4. Let $H_3 : \{0,1\}^* \to \{-1,0,1\}^{n\times n}$ be a secure hash function, and every column vector in the image has a small Hamming weight bounded by λ_2.

5. Let $H_4 : \mathbb{Z}_q^n \to \{0,1\}^{l_1+l_2}$ be a secure hash function, where l_2 is also the length of message ϖ.

6. Let $F_1 : \{0,1\}^{l_2} \to \{0,1\}^{l_1}$, $F_2 : \{0,1\}^{l_1} \to \{0,1\}^{l_2}$ be encoding functions.

Finally, PKG outputs public parameters $params{=}(A, H_1, H_2, H_3, H_4, H_5, F_1, F_2)$ and the secret master key $msk = T$.

- *KeyExtract* (msk, id): Given an identity $id \in \{0,1\}^*$, PKG works as follows:

  1. Sample $E_{id} \leftarrow \mathbb{D}_{\sigma}^{n \times n}$, such that $|E_{id}(i,j)| \leq 7\sigma$ for all $i, j = 1, \cdots, n$. If $|E_{id}(i,j)| > 7\sigma$ for some i, j, Resample again. According to [22], the probability of $|E_{id}(i,j)| > 7\sigma$ for some i, j is less than $1/30$.

  2. Invoke algorithm $SamplePre(A, T, H_1(id) - E_{id}, \sigma)$, provide S_{id} follows the distribution $\mathbb{D}_{\sigma}^{m \times n}$, such that $AS_{id} = H_1(id) - E_{id}$.

  3. Return $sk_{id} = S_{id}$ as secret key for the identity id.
- *DelGen* $\left(sk_{id_O}, id_P, w\right)$: The original signer id_O inputs his secret key $sk_{id_O} = S_{id_O}$, and the warrant $w \in \{0,1\}^*$ associated with proxy signer id_P does the following steps:

  1. Sample $y_w \leftarrow U(D_B^m)$, $U(D_B^m)$ is the uniform distribution on $D_B = [-B, B]$.

  2. Let $c_w = H_2\left(Ay_w \left(\mathrm{mod} q\right)_d, w\right)$, $z_w = S_{id_O} \cdot c_w + y_w$.

  3. Let $\omega = Az_w - H_1\left(id_O\right) \cdot c_w \left(\mathrm{mod} q\right)$. If $\left|\left[\omega_{(i)}\right]_{2^d}\right| > 2^{d-1} - 7\lambda_1\sigma$, go to the first step to resample y_w.

  4. Return $W_{O \to P} = (z_w, c_w)$ with probability $\min\left(D_B^m(z_w) / \left(M \cdot \mathbb{D}_{B, S_{id_O} \cdot c_w}^m(z_w)\right), 1\right)$, and publish delegation information $d_g = (id_O, id_P, w, W_{O \to P} = (z_w, c_w))$ to all users.
- *DelVer* $\left(d_g = (id_O, id_P, w, W_{O \to P} = (z_w, c_w))\right)$: For arbitrary users, he verifies the legality of delegation information $d_g = (id_O, id_P, w, W_{O \to P} = (z_w, c_w))$ as follows:

  1. Compute $\omega = Az_w - H_1\left(id_O\right) \cdot c_w \left(\mathrm{mod} q\right)$.

  2. If $c_w = H_2\left(\lfloor \omega \rceil_d, w\right)$ and $\|z_w\|_{\infty} \leq B$, output 1 and accept this delegation. Otherwise, output 0 and reject it.
- *PkeyGen* $\left(sk_{id_P}, d_g = (id_O, id_P, w, W_{O \to P} = (z_w, c_w))\right)$: the proxy signer id_P inputs his secret key $sk_{id_P} = S_{id_P}$ and the delegation information $d_g = (id_O, id_P, w, W_{O \to P} = (z_w, c_w))$, computes $L_w = H_3(w, z_w, c_w) \in \{-1, 0, 1\}^{n \times n}$, outputs $sk_{O,P,w} = S_{id_P} \cdot L_w \in \mathbb{D}_{\sigma \cdot \sqrt{\lambda_2}}^{m \times n}$ as the delegated secret key.
- *PSign*$(sk_{O,P,w}, \varpi)$: the proxy signer id_P inputs his delegated secret key $sk_{O,P,w} = S_{id_P} \cdot L_w$, the message $\varpi \in \{0,1\}^{l_2}$, does the next steps.

  1. Sample $y \leftarrow U(D_B^m)$, compute $c' = H_4\left(\lfloor Ay \left(\mathrm{mod} q\right) \rceil_d\right)$.

  2. Let $\varpi' = F_1(\varpi) \,||\, (F_2(F_1(\varpi)) \oplus \varpi)$, $c = c' \oplus \varpi'$.

  3. Compute $c_0 = H_5(c)$, $z = S_{id_P} \cdot L_w \cdot c_0 + y$.

  4. Let $\omega = Az - H_1\left(id_P\right) \cdot L_w \cdot c_0 \left(\mathrm{mod} q\right)$.

  5. If $\left|\left[\omega_{(i)}\right]_{2^d}\right| > 2^{d-1} - 7\lambda_1\sqrt{\lambda_2}\sigma$, go to the first step to resample y. Otherwise, return proxy signature $\varsigma = (z, c)$ with probability $\min\left(D_B^m(z) / \left(M \cdot \mathbb{D}_{B, S_{id_P} L_w c_0}^m(z)\right), 1\right)$
- *PVer* $\left(d_g = (id_O, id_P, w, W_{O \to P} = (z_w, c_w)), \varsigma = (z, c)\right)$: For arbitrary user, he verifies the proxy signature with the next steps. Here, we think the legality of delegation information $d_g = (id_O, id_P, w, W_{O \to P} = (z_w, c_w))$ has already been verified.

  1. Compute $c' = H_4(\lfloor Az - H_1\left(id_P\right) \cdot L_w \cdot H_5(c) \left(\mathrm{mod} q\right)_d \rceil)$.

  2. Compute $\varpi' = c \oplus c'$, $\varpi = |\varpi'|_{l_2} \oplus F_2\left(|\varpi'|^{l_1}\right)$.

  3. If $F_1(\varpi) = |\varpi'|^{l_1}$ and $\|z\|_{\infty} < B$, accept the signature and output 1; otherwise, output 0 and reject the signature.

5. Scheme Analysis

5.1. Parameter Setting

n is the system security parameter:

1. For the $TrapGen(q,m)$ algorithm, $q = poly(n)$, $m = \lceil 6n\log q \rceil$.
2. For the $SamplePre(A, T, H_1(id) - E_{id}, \sigma)$ algorithm, $\sigma = \omega\left((m\log m)^{1/2}\right)$.
3. According to [22], λ_1 satisfies $2^{\lambda_1} \cdot \binom{n}{\lambda_1} \geq 2^{128}$.
4. According to [23], l_1 and l_2 are all about 100.
5. According to [22], $2^d \geq 7\lambda_1\sqrt{\lambda_2}n\sigma$, $B = 14\sigma(m-1)\sqrt{\lambda_1\sqrt{\lambda_2}}$.
6. According to [21], M is a small constant of about 8.

5.2. Correctness of the Scheme

1. For $DelVer(d_g = (id_O, id_P, w, W_{O\to P}))$ algorithm, $W_{O\to P} = (z_w, c_w)$,

$$\begin{aligned}\omega &= Az_w - H_1(id_O)\cdot c_w \,(\mathrm{mod}\, q) = A\left(S_{id_O}\cdot c_w + y_w\right) - \\ &\left(AS_{id_O} + E_{id_O}\right)\cdot c_w \,(\mathrm{mod}\, q) \\ &= AS_{id_O}\cdot c_w + Ay_w - AS_{id_O}\cdot c_w - E_{id_O}\cdot c_w \,(\mathrm{mod}\, q) \\ &= Ay_w - E_{id_O}c_w \,(\mathrm{mod}\, q).\end{aligned}$$

Because in step 3 of $DelGen\left(sk_{id_O}, id_P, w\right)$ algorithm, we have:
If $\left|\left[\omega_{(i)}\right]_{2^d}\right| > 2^{d-1} - 7\lambda_1\sigma$, go to the first step to resample y_w.
Therefore, $\lfloor\omega_d\rceil = \lfloor Ay_w - E_{id_O}c_w \,(\mathrm{mod}\, q)\rceil_d = Ay_w \,(\mathrm{mod}\, q)_d$, such that
$c_w = H_2(\lfloor Ay_w(\mathrm{mod}\, q)\rceil_d, w) = H_2(\lfloor\omega_d\rceil, w)$.
In addition, due to $y_w \leftarrow D_B^m$, and $z_w = S_{id_O}\cdot c_w + y_w$, z_w follows uniform distribution on $[-B+\gamma, B-\gamma]^m$ for $\gamma = 14\sqrt{\lambda_1}\sigma$, so that $\|z_w\|_\infty \leq B$. So far, verification of delegation information is correct.

2. For $PVer(d_g = (id_O, id_p, w, W_{O\to P}).\varsigma = (z, c))$ algorithm,

$$\begin{aligned}\omega &= Az - H_1(id_P)\cdot L_w\cdot H_5(c)\,(\mathrm{mod}\, q), \\ &= A\left(S_{id_P}\cdot L_w\cdot H_5(c) + y\right) - \left(AS_{id_P} + E_{id_P}\right)\cdot L_w\cdot H_5(c)\,(\mathrm{mod}\, q), \\ &= Ay - E_{id_P}\cdot L_w\cdot H_5(c)\,(\mathrm{mod}\, q).\end{aligned}$$

Because in step 5 of $PSign(sk_{O,P,w}, \varpi)$ algorithm, we have:
If $\left|\left[\omega_{(i)}\right]_{2^d}\right| > 2^{d-1} - 7\lambda_1\sqrt{\lambda_2}\sigma$, go to the first step to resample y.
Therefore, $\lfloor\omega\rceil_d = \lfloor Ay - E_{id_P}\cdot L_w\cdot H_5(c)(\mathrm{mod}\, q)\rceil_d = \lfloor Ay(\mathrm{mod}\, q)\rceil_d$, such that
$c' = H_4(\lfloor Az - H_1(id_P)\cdot L_w\cdot H_5(c)(\mathrm{mod}\, q)\rceil_d) = H_4(\lfloor Ay(\mathrm{mod}\, q)\rceil_d)$
Due to $c = c' \oplus \varpi'$, we have $\varpi' = c \oplus c'$. Since $\varpi' = F_1(\varpi)\,||\,(F_2(F_1(\varpi)) \oplus \varpi)$, the message $\varpi = |\varpi'|_{l_2} \oplus F_2\left(|\varpi'|^{l_1}\right)$, and $F_1(\varpi) = |\varpi'|^{l_1}$.
In addition, since $y \leftarrow D_B^m$, and $z = S_{id_P}\cdot L_w\cdot H_5(c) + y$, z follows uniform distribution on $[-B+\gamma, B-\gamma]^m$ for $\gamma = 14\sqrt{\lambda_1\lambda_2}\sigma$, so that $\|z\|_\infty \leq B$.
Up to now, proxy signature verification is successful. Combining two points, we draw a conclusion that our scheme is correct.

5.3. Security Analysis

Our scheme security consists of two parts: EUF-ID-CWA security aims at delegation information reliability, EUF-ID-CMA security aims at proxy signature reliability.

5.3.1. EUF-ID-CWA Security

Theorem 1. *Provided that the SIS problem is hard to solve, our identity-based proxy signature scheme with message recovery (IDPSWM) is delegation information existentially unforgeable against adaptive chosen warrant and identity (EUF-ID-CWA).*

Proof. We prove this theorem by contradiction. Assuming that a polynomial time forger $\mathcal{F}$ has the ability to provide valid and fresh delegation information with some non-negligible probability ε_1, we can design an algorithm to solve an SIS instance with probability

$$\left({}^{1}/_{2}-{}^{1}/_{2^{128}}\right)\left(\varepsilon_1-{}^{1}/_{2^{128}}\right)\left(\left(\varepsilon_1-{}^{1}/_{2^{128}}\right)/(Q_1+Q_2)\ -{}^{1}/_{2^{128}}\right),$$

where Q_1 and Q_2 are the times of $H_2(w_{ij})$ queries and $DelGen\left(id_i, id_j, w_{ij}\right)$ queries.

That is to say, with an SIS problem instance $(A|I_n) \in \mathbb{Z}_q^{n\times(m+n)}$, $\mathcal{C}$ interacts with forger $\mathcal{F}$ to find small non-zero vector $e = \begin{pmatrix} e_1 \\ e_2 \end{pmatrix}$, $e_1 \in \mathbb{Z}^m$ and $e_2 \in \mathbb{Z}^n$, such that $(A|I_n)\, e = (A|I_n) \begin{pmatrix} e_1 \\ e_2 \end{pmatrix} = Ae_1 + e_2 = 0 \,(\mathrm{mod} q)$. The details are as follows:

- Initial Phase: $\mathcal{C}$ selects $F_1 : \{0,1\}^{l_2} \to \{0,1\}^{l_1}$, $F_2 : \{0,1\}^{l_1} \to \{0,1\}^{l_2}$, submits A, F_1, and F_2 as system parameters to the forger $\mathcal{F}$.
- Query Phase: The forger $\mathcal{F}$ makes the following queries, $\mathcal{C}$ gives reasonable answers:

  1. $H_1\left(id_i\right)$ query: $\mathcal{F}$ selects a user identity $id_i \in \{0,1\}^*$, sends it to $\mathcal{C}$. $\mathcal{C}$ samples $S_{id_i} \leftarrow \mathbb{D}_\sigma^{m\times n}$, $E_{id_i} \leftarrow \mathbb{D}_\sigma^{n\times n}$, let $H_1\left(id_i\right) = AS_{id_i} + E_{id_i}$. He saves $\left(id_i, S_{id_i}, AS_{id_i} + E_{id_i}\right)$ in the list $\mathcal{H}_1$ and returns $H_1\left(id_i\right) = AS_{id_i} + E_{id_i}$ to $\mathcal{F}$.

  2. $H_2\left(w_{ij}\right)$ query: $\mathcal{F}$ selects warrant $w_{ij} \in \{0,1\}^*$ associated with the original signer $id_i \in \{0,1\}^*$, the proxy signer $id_j \in \{0,1\}^*$, sends all of them to $\mathcal{C}$. $\mathcal{C}$ randomly samples $c_{ij} \leftarrow \{-1,0,1\}^n$ with Hamming weight less than or equal to λ_1, selects $z_{ij} \leftarrow D_B^m$ uniformly, let $\omega = Az_{ij} - H_1\left(id_i\right) \cdot c_{ij} \,(\mathrm{mod} q)$. If some entry in ω is larger than $2^{d-1} - 7\lambda_1\sigma$, $\mathcal{C}$ resamples c_{ij} and z_{ij} again. Because $2^d \geq 7\lambda_1\sqrt{\lambda_2}n\sigma$, the probability that every entry in ω is smaller than $2^{d-1} - 7\lambda_1\sigma$ is larger than $1/3$. At last, $\mathcal{C}$ saves $\left(id_i, id_j, w_{ij}, c_{ij}, z_{ij}\right)$ in list $\mathcal{H}_2$ and returns c_{ij} to $\mathcal{F}$.

  3. $KeyExtract\left(id_i\right)$ query: $\mathcal{F}$ selects a user identity $id_i \in \{o,1\}^*$ and sends it to the challenger $\mathcal{C}$. $\mathcal{C}$ searches list $\mathcal{H}_1$ to get $(id_i, D_{id_i}, AS_{id_i} + E_{id_i})$, and returns $sk_{id_i} = S_{id_i}$. If it doesn't exist, $\mathcal{C}$ queries $H_1\left(id_i\right)$ firstly.

  4. $DelGen\left(id_i, id_j, w_{ij}\right)$ query: $\mathcal{F}$ selects the original signer $id_i \in \{0,1\}^*$, the proxy signer $id_j \in \{0,1\}^*$, and the warrant $w_{ij} \in \{0,1\}^*$, sends all of them to $\mathcal{C}$. $\mathcal{C}$ looks list H_2 for $(id_i, id_j, w_{ij}, c_{ij}, z_{ij})$ and returns $\left(z_{ij}, c_{ij}\right)$. If $(id_i, id_j, w_{ij}, c_{ij}, z_{ij})$ doesn't exist, $\mathcal{C}$ queries $H_2\left(w_{ij}\right)$ firstly.
- Forge Phase: The forger $\mathcal{F}$ gives his forgery $(id_{i^*}, id_{j^*}, w_{ij^*}, W_{i^*\to j^*} = (z^*, c^*))$.

  Because $\mathcal{F}$ queries $H_2(w_{ij})$ at most Q_1 times, queries $DelGen\left(id_i, id_j, w_{ij}\right)$ at most Q_2 times, so that the number of c_{ij} is at most $Q_1 + Q_2$. Suppose there are c_1, c_2, $\cdots$, $c_{Q_1+Q_2}$. For $Az^* - H_1\left(id_{i^*}\right) c^* \,(\mathrm{mod} q)$, the probability of $\mathcal{F}$ generates c^* such that $c^* = H_2\left(\lfloor Az^* - H_1\left(id_{i^*}\right) \cdot c^* \,(\mathrm{mod} q)\rceil_d, w_{ij^*}\right)$ is $1/\left(2^{128}\right)$, which is negligible, so that $c^* \in \left\{c_1, c_2, \cdots, c_{Q_1+Q_2}\right\}$ with overwhelming probability $1 - 1/\left(2^{128}\right)$.

Because $\mathcal{F}$ gives a successful forgery with probability ε_1, $(id_{i^*}, id_{j^*}, w_{ij^*}, W_{i^*\to j^*} = (z^*, c^*))$ is a valid forgery and $c^* \in \left\{c_1, c_2, \cdots, c_{Q_1+Q_2}\right\}$ with probability $\varepsilon_1 - 1/2^{128}$. Supposing $c^* = c_t$, we further conclude that it comes from a H_2 query rather than a *DelGen* query.

If $c^* = c_t$ comes from $DelGen\left(id_{i_t}, id_{j_t}, w_{ij_t}\right)$ query, then

$$c^* = H_2\left(\lfloor Az^* - H_1\left(id_{i^*}\right) \cdot c^* \left(\mathrm{mod} q\right)\rceil_d, w_{ij^*}\right) = H_2\left(\lfloor Az_t - H_1\left(id_{i_t}\right) \cdot c^* \left(\mathrm{mod} q\right)\rceil_d, w_{ij_t}\right).$$

If $w_{ij^*} \neq w_{ij_t}$ or $\lfloor Az^* - H_1\left(id_{i^*}\right) \cdot c^* \left(\mathrm{mod} q\right)\rceil_d \neq \lfloor Az_t - H_1\left(id_{i_t}\right) \cdot c^* \left(\mathrm{mod} q\right)\rceil_d$, then a collision in H_2 is obtained.

Therefore, $w_{ij^*} = w_{ij_t}$, which leads to $\left(id_{i^*}, id_{j^*}, w_{ij^*}\right) = \left(id_{i_t}, id_{j_t}, w_{ij_t}\right)$(because the warrant includes the identity information), and the entries of $A\left(z^* - z_t\right)\left(\mathrm{mod} q\right)$ are in $\left[-2^d, 2^d\right]$.

If $z^* = z_t$, $\left(id_{i^*}, id_{j^*}, w_{ij^*}, W_{i^* \to j^*} = (z^*, c^*)\right) = \left(id_{i_t}, id_{j_t}, w_{ij_t}, W_{i_t \to j_t} = (z_t, c_t)\right)$, it isn't a successful forgery.

If $z^* \neq z_t$, let $e_1 = z^* - z_t$, $e_2 = -A\left(z^* - z_t\right)\left(\mathrm{mod} q\right)$, then $Ae_1 + e_2 = 0\left(\mathrm{mod} q\right)$, and $\|e_1\|_\infty \leq 2B$, $\|e_2\|_\infty \leq 2^d$. The SIS instance is solved.

Now, we know $c^* = c_t$ comes from $H_2(w_{ij})$ query, and invoke $\mathcal{F}$ again. Due to General Forking Lemma [29], with a probability not less than

$$\left(\varepsilon_1 - {}^1/_{2^{128}}\right)\left(\left(\varepsilon_1 - {}^1/_{2^{128}}\right)/\left(Q_1 + Q_2\right) - {}^1/_{2^{128}}\right),$$

we obtain a different valid delegation information $(\bar{z}, \bar{c})$ on $\left(id_{i^*}, id_{j^*}, w_{ij^*}\right)$, and $\bar{c} \neq c^*$.

Then, $\lfloor Az^* - H_1\left(id_{i^*}\right) \cdot c^* \left(\mathrm{mod} q\right)\rceil_d = \lfloor A\bar{z} - H_1\left(id_{i^*}\right) \cdot \bar{c}\left(\mathrm{mod} q\right)\rceil_d$, which means $Az^* - H_1\left(id_{i^*}\right) \cdot c^* + e = A\bar{z} - H_1\left(id_{i^*}\right) \cdot \bar{c}\left(\mathrm{mod} q\right)$ for $\|e\|_\infty \leq 2^{d-1}$. Replacing $H_1\left(id_{i^*}\right)$ with $AS_{id_{i^*}} + E_{id_{i^*}}$, we have $A\left(z^* - \bar{z} + S_{id_{i^*}}\left(\bar{c} - c^*\right)\right) + e + E_{id_{i^*}}\left(\bar{c} - c^*\right) = 0\left(\mathrm{mod} q\right)$. Let $e_1 = z^* - \bar{z} + S_{id_{i^*}}\left(\bar{c} - c^*\right)$, $e_2 = e + E_{id_{i^*}}\left(\bar{c} - c^*\right)$, then $\|e_1\|_\infty \leq 2B + 2\lambda_1\sigma$, $\|e_2\|_\infty \leq 2^{d-1} + 2\lambda_1\sigma$. In addition, $S_{id_{i^*}}$ and $E_{id_{i^*}}$ have a variety of options, $\mathcal{F}$ doesn't know which pair $\left(S_{id_{i^*}}, E_{id_{i^*}}\right)$ is used to build e_1 and e_2. Therefore, the probability of $(e_1, e_2) \neq (0, 0)$ is at least $1/2$. □

5.3.2. EUF-ID-CMA Security

Theorem 2. *Provided that the SIS problem is hard to solve, our identity-based proxy signature scheme with message recovery (IDPSWM) is signature existentially unforgeable against adaptive chosen message and identity (EUF-ID-CMA).*

Proof. We prove this theorem by contradiction. Assuming that a polynomial time forger $\mathcal{F}$ has the ability to provide a valid and fresh proxy signature with some non-negligible probability ε_2, we can design an algorithm $\mathcal{C}$ to solve an SIS problem instance with probability

$$\left({}^1/_2 - {}^1/_{2^{128}}\right)\left(\varepsilon_2 - {}^1/_{2^{128}}\right)\left(\left(\varepsilon_2 - {}^1/_{2^{128}}\right)/\left(Q_3 + Q_4\right) - {}^1/_{2^{128}}\right),$$

where Q_3 and Q_4 are the times of $H_5\left(c\right)$ queries and $PSign\left(\left(id_i, id_j, w_{ij}, c_{ij}, z_{ij}\right), \varpi_k\right)$ queries.

That is to say, with an SIS problem instance $\left(A|I_n\right) \in \mathbb{Z}_q^{n \times (m+n)}$, $\mathcal{C}$ interacts with forger $\mathcal{F}$ to find a small non-zero vector $e = \begin{pmatrix} e_1 \\ e_2 \end{pmatrix}$, $e_1 \in \mathbb{Z}^m$ and $e_2 \in \mathbb{Z}^n$, such that $\left(A|I_n\right)e = \left(A|I_n\right)\begin{pmatrix} e_1 \\ e_2 \end{pmatrix} = Ae_1 + e_2 = 0\left(\mathrm{mod} q\right)$. The details are as follows:

- Initial Phase: $\mathcal{C}$ selects $F_1 : \{0,1\}^{l_2} \to \{0,1\}^{l_1}$, $F_2 : \{0,1\}^{l_1} \to \{0,1\}^{l_2}$, submits A, F_1, and F_2 as system parameters to the forger $\mathcal{F}$.
- Query Phase: The forger $\mathcal{F}$ makes the following queries, $\mathcal{C}$ gives reasonable answers:

  1. $H_1\left(id_i\right)$ query: $\mathcal{F}$ selects a user identity $id_i \in \{0,1\}^*$, and sends it to $\mathcal{C}$. $\mathcal{C}$ samples $S_{id_i} \leftarrow \mathbb{D}_\sigma^{m \times n}$, $E_{id_i} \leftarrow \mathbb{D}_\sigma^{n \times n}$, let $H_1\left(id_i\right) = AS_{id_i} + E_{id_i}$. He saves $\left(id_i, S_{id_i}, AS_{id_i} + E_{id_i}\right)$ in the list $\mathcal{H}_1$ and returns $H_1(id_i) = AS_{id_i} + E_{id_i}$ to $\mathcal{F}$.

2. $H_2\left(w_{ij}\right)$ query: $\mathcal{F}$ selects warrant $w_{ij} \in \{0,1\}^*$ associated with the original signer $id_i \in \{0,1\}^*$, the proxy signer $id_j \in \{0,1\}^*$, sends all of them to $\mathcal{C}$. $\mathcal{C}$ randomly samples $c_{ij} \leftarrow \{-1,0,1\}^n$ with Hamming weight less than or equal to λ_1, selects $z_{ij} \leftarrow D_B^m$ uniformly, let $\omega = Az_{ij} - H_1\left(id_i\right) \cdot c_{ij} \left(\mathrm{mod} q\right)$. If some entry in ω is larger than $2^{d-1} - 7\lambda_1\sigma$, $\mathcal{C}$ resamples c_{ij} and z_{ij} again. Because $2^d \geq 7\lambda_1\sqrt{\lambda_2}n\delta$, the probability that every entry in ω is smaller than $2^{d-1} - 7\lambda_1\sigma$ is larger than $1/3$. At last, $\mathcal{C}$ saves $\left(id_i, id_j, w_{ij}, c_{ij}, z_{ij}\right)$ in list $\mathcal{H}_2$ and returns c_{ij} to $\mathcal{F}$.

3. $H_4\left(y\right)$ query: $\mathcal{F}$ selects $y \leftarrow U\left(D_B^m\right)$ randomly, sends it to $\mathcal{C}$. $\mathcal{C}$ selects $c' \in \{0,1\}^{l_1+l_2}$ uniformly and randomly. Then, $\mathcal{C}$ saves $\left(y, \lfloor Ay \left(\mathrm{mod} q\right) \rceil_d, c'\right)$ in list $\mathcal{H}_4$ and returns c' to $\mathcal{F}$.

4.$H_5\left(c\right)$ query: $\mathcal{F}$ sends $c \in \{0,1\}^{l_1+l_2}$and submits it to $\mathcal{C}$. $\mathcal{C}$ chooses $c_0 \leftarrow \{-1,0,1\}^n$ with Hamming weight less than or equal to λ_1. Then, $\mathcal{C}$ saves $\left(c, c_0\right)$ in list $\mathcal{H}_5$ and returns c_0 to $\mathcal{F}$.

5. *KeyExtract* $\left(id_i\right)$ query: $\mathcal{F}$ selects a user identity $id_i \in \{0,1\}^*$, sends it to the challenger $\mathcal{C}$. $\mathcal{C}$ searches list $\mathcal{H}_1$ to get $\left(id_i, S_{id_i}, AS_{id_i} + E_{id_i}\right)$, returns $sk_{id_i} = S_{id_i}$. If it doesn't exist, $\mathcal{C}$ queries $H_1\left(id_i\right)$ firstly.

6. *DelGen* $\left(id_i, id_j, w_{ij}\right)$ query: $\mathcal{F}$ selects the original signer $id_i \in \{0,1\}^*$, the proxy signer $id_j \in \{0,1\}^*$, and the warrant $w_{ij} \in \{0,1\}^*$ sends all of them to $\mathcal{C}$. $\mathcal{C}$ looks list H_2 for $\left(id_i, id_j, w_{ij}, c_{ij}, z_{ij}\right)$ and returns $\left(z_{ij}, c_{ij}\right)$. If $\left(id_i, id_j, w_{ij}, c_{ij}, z_{ij}\right)$ doesn't exist, $\mathcal{C}$ queries $H_2\left(w_{ij}\right)$ firstly.

7. *PkeyGen* $\left(id_i, id_j, w_{ij}, z_{ij}, c_{ij}\right)$ query: $\mathcal{F}$ sends the delegation information $\left(id_i, id_j, w_{ij}, z_{ij}, c_{ij}\right)$ to the challenger $\mathcal{C}$. $\mathcal{C}$ verifies its validity firstly. If it isn't valid, he refuses to respond. Otherwise, $\mathcal{C}$ executes *KeyExtract* $\left(id_j\right)$ query to get secret key $sk_{id_j} = S_{id_j}$, computes $L_{w_{ij}} = H_3\left(w_{ij}, z_{ij}, c_{ij}\right)$ and $sk_{i,j,w_{ij}} = S_{id_j} \cdot L_{w_{ij}}$, returns $sk_{i,j,w_{ij}}$ to $\mathcal{F}$.

8. *PSign* $\left(\left(id_i, id_j, w_{ij}, z_{ij}, c_{ij}\right), \varpi_k\right)$ query: $\mathcal{F}$ submits $\left(id_i, id_j, w_{ij}, z_{ij}, c_{ij}\right)$ and message ϖ_k to the challenger $\mathcal{C}$. $\mathcal{C}$ verifies the legality of $\left(id_i, id_j, w_{ij}, z_{ij}, c_{ij}\right)$. If it is illegal, $\mathcal{C}$ rejects answering the query. Otherwise, he executes *PkeyGen* $\left(id_i, id_j, w_{ij}, z_{ij}, c_{ij}\right)$ query to get the delegated secret key $sk_{i,j,w_{ij}}$, invokes algorithm *PSign* $\left(sk_{i,j,w_{ij}}, \varpi_k\right)$ to get signature $\varsigma_{ijk} = \left(z_{ijk}, c_{ijk}\right)$, and returns it to $\mathcal{F}$.

- Forge Phase: The forger $\mathcal{F}$ gives his forgery signature $\left(id_{i^*}, id_{j^*}, w_{ij^*}, z^*, c^*, z_{\varpi^*}, c_{\varpi^*}\right)$ for message ϖ^*.

  $\mathcal{C}$ invokes $\mathcal{F}$ again. Due to General Forking Lemma [29], with probability $\left(\varepsilon_2 - {}^1/_{2^{128}}\right)\left(\left(\varepsilon_2 - {}^1/_{2^{128}}\right)/\left(Q_3 + Q_4\right) - {}^1/_{2^{128}}\right)$, we obtain a new signature $\left(id_{i^*}, id_{j^*}, w_{ij^*}, z^*, c^*, z'_{\varpi^*}, c'_{\varpi^*}\right)$ for message ϖ^*, such that

$$\lfloor Az_{\varpi^*} - H_1\left(id_{j^*}\right) \cdot L_{w_{ij^*}} \cdot H_5\left(c_{\varpi^*}\right) \left(\mathrm{mod} q\right) \rceil_d$$

  is equivalent to

$$\lfloor Az'_{\varpi^*} - H_1\left(id_{j^*}\right) \cdot L_{w_{ij^*}} \cdot H_5\left(c'_{\varpi^*}\right) \left(\mathrm{mod} q\right) \rceil_d$$

  and $H_5\left(c'_{\varpi^*}\right) \neq H_5\left(c_{\varpi^*}\right)$.

  Then, $Az_{\varpi^*} - H_1\left(id_{j^*}\right) \cdot L_{w_{ij^*}} \cdot H_5\left(c_{\varpi^*}\right) + \hat{e} = Az'_{\varpi^*} - H_1\left(id_{j^*}\right) \cdot L_{w_{ij^*}} \cdot H_5\left(c'_{\varpi^*}\right) \left(\mathrm{mod} q\right)$ for $\|\hat{e}\|_\infty \leq 2^{d-1}$. Replacing $H_1\left(id_{j^*}\right)$ with $AS_{id_{j^*}} + E_{id_{j^*}}$, we have $A\left(z_{\varpi^*} - z'_{\varpi^*} + S_{id_{j^*}} \cdot L_{w_{ij^*}}\left(H_5\left(c'_{\varpi^*}\right) - H_5\left(c_{\varpi^*}\right)\right)\right) + \hat{e} + E_{id_{j^*}}\left(H_5\left(c'_{\varpi^*}\right) - H_5\left(c_{\varpi^*}\right)\right) = 0 \left(\mathrm{mod} q\right)$. Let $e_1 = z_{\varpi^*} - z'_{\varpi^*} + S_{id_{j^*}} \cdot L_{w_{ij^*}}\left(H_5\left(c'_{\varpi^*}\right) - H_5\left(c_{\varpi^*}\right)\right)$, $e_2 = \hat{e} + E_{id_{j^*}}\left(H_5\left(c'_{\varpi^*}\right) - H_5\left(c_{\varpi^*}\right)\right)$, then $\|e_1\|_\infty \leq 2B + 2\lambda_1\sqrt{\lambda_2}\sigma$, $\|e_2\|_\infty \leq 2^{d-1} + 2\lambda_1\sigma$. In addition, $S_{id_{i^*}}$ and $E_{id_{i^*}}$ have a variety of options, $\mathcal{F}$ doesn't know which pair $\left(S_{id_{j^*}}, E_{id_{j^*}}\right)$ is used to build e_1 and e_2. Therefore, the probability of $\left(e_1, e_2\right) \neq (0,0)$ is at least $1/2$.

□

5.4. Performance Analysis

Regarding the performance analysis, we will focus on the following three aspects: signature compression, signing right delegation and message recovery.

Firstly, we take the signature compression technique from [22]. For hash value $H_1(id)$ for user id, we first sample $E_{id} \leftarrow \mathbb{D}_{\sigma}^{n \times n}$ such that $|E_{id}(i,j)| \leq 7\sigma$ for all $i, j = 1, \cdots, n$. Then, we invoke algorithm $S_{id} \leftarrow SamplePre(A, T, H_1(id) - E_{id}, \sigma)$ such that $AS_{id} + E_{id} = H_1(id)$. We set S_{id} rather than (S_{id}, E_{id}), as the private key of user id. The abandoned E_{id} leads to the signature length reducing from $(S_{id}c + y, E_{id}c + y)$ to $S_{id}c + y$, which is about $n \log\left(14\sigma(m-1)\sqrt{\lambda_1\sqrt{\lambda_2}}\right)$ bits. Combining the operation $\lfloor a \rceil_d = (a - [a]_{2^d})/2^d$, the discarded $E_{id}c + y$ does not affect signature verification algorithm.

For signing right delegation, we make the original signer's signature (z_w, c_w) for the warrant w public for everyone. Any verifier can take (w, z_w, c_w) to verify the original signer's signing right transfer to the proxy signer. Besides doing the same operations with the verifier, the proxy signer must embed (w, z_w, c_w) into the generation of proxy signature private key—the delegated secret key. Therefore, the delegated secret key is decided by the original signer and the proxy signer. The original signer can't deny his authorization to the proxy signer, can't generate the delegated secret key alone, so that proxy signer's interests are protected. On the other hand, the proxy signer can't generate the delegated secret key without the permission of the original signer, thus the interests of the original signer are protected. In addition, no secure channel is necessary between the original signer and proxy signer—because no secret information is transmitted between them.

Thirdly, we use the idea of message recovery signature in [23], hide the message ϖ in the signature, and the message ϖ can be recovered without any secret information, hence only the signature should be transmitted and everyone can verify its legality.

In Table 1, we give the performance comparison between [23] and our scheme. Two schemes are both with message recovery and quantum resistance, and the number of signature verification operations is the same. The differences between two schemes are shown in the following aspects: firstly, the scheme in [23] needs the support of public key infrastructure while our scheme does not need it. Public key infrastructure provides security assurance of the relationship between public key and private key, which is achieved by authoritative authority signing certificates for users. Therefore, public key infrastructure needs to complete certificate allocation, verification, storage and revocation operations, which requires a large amount of bandwidth resources and computing resources. In our scheme, the public key is the user's identity, and the relationship between the public key and the private key is natural. Therefore, we no longer need the support of the complex public key infrastructure, and the system becomes concise. Secondly, the scheme in [23] does not have the function of proxy authorization, and our scheme has this function. Therefore, our scheme is more powerful. In addition, the scheme in [23] does not introduce signature compression technology, and our scheme introduces signature compression technology to make the signature length shorter. It is clear that our scheme has better functionality compared to the scheme in [23]. However, because we take the signature compression technique from [22] to condense signature length, it is necessary to ensure $\left|\left[\omega_{(i)}\right]_{2^d}\right| \leq 2^{d-1} - 7\lambda_1\sqrt{\lambda_2}\sigma$. To this end, we repeat operations of signing message with probability not larger than $2/3$—this is our scheme's extra computation cost. For every operation of signing message and verification, our scheme's computation cost is comparable with that of the scheme in [23].

The lattice-based proxy signature scheme with message recovery in [25] follows the same frame with the scheme in [23] and ours; we also include it in Table 1. Compared with our scheme, the scheme in [25] bases on public key infrastructure, delegation of signature right depends on secure channel and can't be verified publicly. In addition, the scheme in [25] doesn't take signature compression technique, its signature is longer and the number of signature operations is small. According to [30], reduction in message length will reduce energy consumption to a greater extent than reduction in computation. Overall, our scheme is more efficient.

Table 1. Performance comparison among Refs. [23] and [25] and our scheme.

	[23]	[25]	Ours
Public key infrastructure	Need	Need	Not need
Delegation of signature right	No	Yes	Yes
Signature compression	No	No	Yes
Message recovery	Yes	Yes	Yes
Quantum resistance	Yes	Yes	Yes
Signature operation	1 time	1 time	5/3 time
Verification operation	1 time	1 time	1 time

6. Conclusions

In this paper, we first proposed the identity-based proxy signature scheme with message recovery based on the lattice assumptions. In particular, we used the signature compression technique for lattice signature without trapdoors to decrease signature length. We abandoned the secure channel between original signer and proxy signer and made the model possess better environmental adaptability. We also divided the security definition into two factors, making the security analysis much easier to be understood. We introduced the idea of message recovery signature, embedding messages into signatures and shortening the amount of information to be transmitted. For security analysis, our scheme is based on the learning with errors and the small integer solution problems. Finally, we demonstrated our performance via comparison with some related works.

Author Contributions: The first author X.L., proposed the main idea as well as the concrete schemes of the paper. The second and third authors Q.W. and W.Y., gave all the figures and tables, as well as Sections 1 and 2. The fourth author K.L. contributed to the security analysis part. The fifth and sixth authors Z.J. and E.P., were responsible for the English writing of the whole paper. The seventh author J.C., the corresponding author, was responsible for efficiency analysis as well as the management of the research project.

Funding: This work was funded by the National Natural Science Foundation of China (No. 61502044, 61402015, 61702212); the Fundamental Research Funds for the Central Universities (No. 2015RC23); the Natural Science Foundation of Hebei Province (No. F2018408040); the Natural Science Foundation of Shandong Province (No. ZR201702180067); and the Hebei Education Funds for Youth Project (No. QN2018047).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mambo, M.; Usuda, K.; Okamoto, E. Proxy signatures for delegating signing operation. In Proceedings of the 3rd ACM Conference on Computer and Communications Security, New Delhi, India, 14–15 March 1996; pp. 48–57.
2. Wei, J.; Yang, G.; Mu, Y.; Liang, K. Anonymous Proxy Signature with Hierarchical Traceability. *Comput. J.* **2016**, *59*, 559–569. [CrossRef]
3. He, K.; Liu, X.; Yuan, H.; Wei, W.; Liang, K. Hierarchical Conditional Proxy Re-Encryption: A New Insight of Fine-Grained Secure Data Sharing. In *Information Security Practice and Experience, Proceedings of the 13th International Conference, ISPEC 2017, Melbourne, Australia, 13–15 December 2017*; Liu, J.K., Samarati, P., Eds.; Springer: Berlin/Heidelberg, Germany, 2017; Volume 10701, pp. 118–135. [CrossRef]
4. Shao, J.; Lu, R.; Lin, X.; Liang, K. Secure bidirectional proxy re-encryption for cryptographic cloud storage. *Pervasive Mob. Comput.* **2016**, *28*, 113–121. [CrossRef]
5. Liang, K.; Susilo, W.; Liu, J.K.; Wong, D.S. Efficient and Fully CCA Secure Conditional Proxy Re-Encryption from Hierarchical Identity-Based Encryption. *Comput. J.* **2015**, *58*, 2778–2792. [CrossRef]
6. Liang, K.; Chu, C.; Tan, X.; Wong, D.S.; Tang, C.; Zhou, J. Chosen-ciphertext secure multi-hop identity-based conditional proxy re-encryption with constant-size ciphertexts. *Theor. Comput. Sci.* **2014**, *539*, 87–105. [CrossRef]
7. Liang, K.; Au, M.H.; Liu, J.K.; Susilo, W.; Wong, D.S.; Yang, G.; Phuong, T.V.X.; Xie, Q. A DFA-Based Functional Proxy Re-Encryption Scheme for Secure Public Cloud Data Sharing. *IEEE Trans. Inf. Forensics Secur.* **2014**, *9*, 1667–1680. [CrossRef]

8. Liang, K.; Liu, J.K.; Wong, D.S.; Susilo, W. An Efficient Cloud-Based Revocable Identity-Based Proxy Re-encryption Scheme for Public Clouds Data Sharing. In *Computer Security, Proceedings of the ESORICS 2014, 19th European Symposium on Research in Computer Security, Wroclaw, Poland, 7–11 September 2014*; Kutylowski, M., Vaidya, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; Volume 8712, pp. 257–272. [CrossRef]
9. Liang, K.; Fang, L.; Susilo, W.; Wong, D.S. A Ciphertext-Policy Attribute-Based Proxy Re-encryption with Chosen-Ciphertext Security. In Proceedings of the 2013 5th International Conference on Intelligent Networking and Collaborative Systems, Xi'an, China, 9–11 September 2013; pp. 552–559. [CrossRef]
10. Nyberg, K.; Rueppel, R.A. A new signature scheme based on the DSA giving message recovery. In Proceedings of the 1st ACM conference on Computer anD Communications Security, Fairfax, VA, USA, 3–5 November 1993; pp. 58–61.
11. Singh, H.; Verma, G.K. ID-based proxy signature scheme with message recovery. *J. Syst. Softw.* **2012**, *85*, 209–214. [CrossRef]
12. Tiwari, N.; Padhye, S. New proxy signature scheme with message recovery using verifiable self-certified public keys. In Proceedings of the 2011 2nd International Conference on Computer and Communication Technology, Allahabad, India, 15–17 September 2011; pp. 539–544.
13. Xie, Q. Provably Secure Self-certified Multi-proxy Signature with Message Recovery. *J. Netw.* **2012**, *7*, 1616. [CrossRef]
14. Yoon, E.J.; Choi, Y.; Kim, C. New ID-based proxy signature scheme with message recovery. In Proceedings of the International Conference on Grid and Pervasive Computing, Seoul, Korea, 9–11 May 2013; pp. 945–951.
15. Mahmoodi, A.; Mohajery, J.; Salmasizadeh, M. A certificate-based proxy signature with message recovery without bilinear pairing. *Security Commun. Netw.* **2016**, *9*, 4983–4991. [CrossRef]
16. Padhye, S.; Tiwari, N. ECDLP-based certificateless proxy signature scheme with message recovery. *Trans. Emerg. Telecommun. Technol.* **2015**, *26*, 346–354. [CrossRef]
17. Shor, P.W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Rev.* **1999**, *41*, 303–332. [CrossRef]
18. Gentry, C. Fully homomorphic encryption using ideal lattices. In Proceedings of the 41st Annual ACM Symposium on Symposium on Theory of Computing-STOC'09, Washington, DC, USA, 31 May–2 June 2009; Volume 9.
19. Gentry, C.; Peikert, C.; Vaikuntanathan, V. Trapdoors for hard lattices and new cryptographic constructions. In Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing, Victoria, BC, Canada, 17–20 May 2008; pp. 197–206.
20. Micciancio, D.; Peikert, C. Trapdoors for Lattices: Simpler, Tighter, Faster, Smaller. In *Advances in Cryptology, Proceedings of the EUROCRYPT 2012, Cambridge, UK, 15–19 April 2012*; Pointcheval, D., Johansson, T., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 700–718.
21. Lyubashevsky, V. Lattice signatures without trapdoors. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, 15–19 April 2012; pp. 738–755.
22. Bai, S.; Galbraith, S.D. An improved compression technique for signatures based on learning with errors. In Proceedings of the Cryptographers' Track at the RSA Conference, San Francisco, CA, USA, 25–28 February 2014; pp. 28–47.
23. Tian, M.; Huang, L. Lattice-based message recovery signature schemes. *Int. J. Electron. Secur. Digit. Forensics* **2013**, *5*, 257–269. [CrossRef]
24. Li, W. An Identity-Based Proxy Signature Scheme from Lattices in the Standard Model. In Proceedings of the International Conference on Intelligent Networking and Collaborative Systems, Ostrawva, Czech Republic, 7–9 September 2016.
25. Wu, F.; Yao, W.; Zhang, X.; Zheng, Z. An Efficient Lattice-Based Proxy Signature with Message Recovery. In Proceedings of the International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage, Guangzhou, China, 12–15 December 2017; Volume 10656, pp. 321–331.
26. Lindell, Y. Fast Secure Two-Party ECDSA Signing. In *Advances in Cryptology, Proceedingds of the CRYPTO 2017, Barbara, CA, USA, 20–24 August 2017*; Katz, J., Shacham, H., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 613–644.

27. Agrawal, S.; Boneh, D.; Boyen, X. Efficient lattice (H) IBE in the standard model. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, French Riviera, France, 30 May–3 June 2010; pp. 553–572.
28. Applebaum, B.; Cash, D.; Peikert, C.; Sahai, A. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In Proceedings of the CRYPTO 2009, Santa Barbara, CA, USA, 16–20 August 2009; pp. 595–618.
29. Bellare, M.; Neven, G. Multi-signatures in the plain public-Key model and a general forking lemma. In Proceedings of the ACM Conference on Computer and Communications Security, Alexandria, VA, USA, 30 October–3 November 2006; pp. 390–399.
30. Hill, J.; Szewczyk, R.; Woo, A.; Hollar, S.; Culler, D.; Pister, K. System architecture directions for networked sensors. *SIGPLAN Not.* **2000**, *35*, 93–104. [CrossRef]