Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 5401890, 9 pages
https://doi.org/10.1155/2018/5401890

Research Article

An Efficient Identity-Based Proxy Blind Signature for Semioffline Services

Hongfei Zhu,[1] **Yu-an Tan**,[1] **Liehuang Zhu**,[1] **Quanxin Zhang**,[1] **and Yuanzhang Li**[1,2]

[1] *School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China*
[2] *Research Center of Massive Language Information Processing and Cloud Computing Application, Beijing 100081, China*

Correspondence should be addressed to Yuanzhang Li; popular@bit.edu.cn

Received 6 February 2018; Accepted 12 March 2018; Published 9 May 2018

Academic Editor: Ximeng Liu

Fog computing extends the cloud computing to the network edge and allows deploying a new type of semioffline services, which can provide real-time transactions between two entities, while the central cloud server is offline and network edge devices are online. For an e-payment system and e-voting with such feature, proxy blind signature is a cornerstone to protect users' privacy. However, the signature based on number theorem, such as hard mathematical problems on factoring problem, discrete logarithm problem, and bilinear pairings, cannot defeat quantum computers attack. Meanwhile, these schemes need to depend on complex public key infrastructure. Thus, we construct an identity-based proxy blind signature scheme based on number theorem research unit lattice, which can defeat quantum computers attack and does not need to depend on public key infrastructure. The security of the proposed scheme is dependent on Ring-Small Integer Solution problem over number theorem research unit lattice. The proposed scheme meets the properties of blind signature and proxy signature. Then we compare the proposed scheme with other existing proxy blind signature schemes; the result shows that the proposed scheme outperforms ZM scheme except in proxy signer's signature size and can be more secure than TA scheme and MMHP scheme.

1. Introduction

Fog computing was initially introduced by Cisco, which can overcome cloud computing's disadvantages, such as non-real-time service and long delay [1–3]. More specifically, fog computing adds a new layer between cloud server and terminal user [4, 5]; that is, fog servers can be access point, base station, router, or mobile equipment [6–9]. Thus, the semioffline e-payment system can be deployed by utilizing the advantages of fog computing model [10, 11].

In order to defend user's privacy in offline e-payment system, blind signature (BS) is crucial for that it never permits signer to sign on a plaintext before knowing its content [12]. Therefore, BS can protect user privacy during the transactions [13] instead of encrypting the data and searching on the ciphertext [14]. However, this system is deployed in real environment; it will use distributed architecture [15]. The original signer should authorize an agent to sign for himself. Then a proxy signature (PS) should be used in e-payment system, since proxy signer can satisfy this requirement [16]. Combining those two types of schemes together, a new proxy blind signature (PBS) was proposed, which meets the properties of those two signature schemes. After that, many PBS schemes were constructed by scholars.

However, most of the PBS schemes are based on number theory, such as discrete logarithm problem (DLP) and bilinear pairings. These schemes are considered to be insecure to resist the quantum computer attack. Therefore, the e-payment and e-voting systems in the cloud still face the threat from quantum computer attack [17]. Meanwhile, these schemes need to rely on complex public key infrastructure (PKI) [18, 19]. In conclusion, these schemes based on number theorem cannot defeat the quantum computers attack according to the recent research results.

Therefore, the lattice-based PBS schemes become one alternative solution, since they are sufficient enough and able to resist quantum computer attack [20, 21]. Besides, if lattice-based PBS schemes can combine with identity-based cryptography (called IDPBS), they can overcome the shortcomings of traditional PBS schemes, such as relying on complex PKI [22]. Meanwhile, they can transfer less data than biological recognition methods during the transactions [23, 24].

Zhu et al. presented a new lattice-based BS [20], which can be secure enough for cloud services. However, this scheme has to be combined with proxy signature in practice. Combining BS scheme and IDPS scheme, we initially present an IDPBS on number theorem research unit lattice (IDPBS-NTRU), which can defeat quantum computer attack.

(1) Inspired by [25], a new IDPBS-NTRU scheme is proposed based on NTRU lattice, which can make semioffline e-payment and e-voting systems deployed in fog computing model secure enough to resist quantum computer attack.

(2) The proposed IDPBS-NTRU scheme is proven to be secure. That is, the proposed scheme is correct, blind, unforgeable, verifiable, strong identifiable, strong undeniable, and key-dependent.

(3) The proposed IDPBS-NTRU is compared with the existing IDPBS schemes in terms of performances. The result shows that it outperforms the ZM scheme except in proxy signer's signing key size, and it is more secure than TA and MMHP schemes.

The paper is introduced as follows. Section 2 introduces the background knowledge about NTRU lattice and main key technology. Section 3 introduces the security model for IDPBS. Section 4 shows that the proposed IDPBS is proven to be secure and it is compared with other IDPBS schemes in terms of performances. At last, Section 5 draws the conclusions.

2. Related Works

IDPBS Schemes Based on DLP. In 2011, Beura et al. proposed a new proxy blind signature based on DLP; their scheme satisfies the properties of blind signature and proxy signature. This scheme is more secure and efficient than factoring signature schemes [26]. To improve the efficiency, Tan et al. introduced a couple of PBS schemes; both of them were constructed on Schnorr blind signature. However, Sun et al. pointed that both of them were not unforgeable and unlinkable [27]. However, in 2014, Wang and Liao proved that the schemes proposed by Oo et al. and Beura et al. did not satisfy unlinkability [28]. In 2013, Tan proposed a PBS based on DLP, which did not depend on PKI [29] and was proven to be secure in the random oracle [30]. However, most of these schemes are dependent on PKI and are not strictly proven to be secure.

IDPBS Schemes Based on Bilinear Pairings. In 2003, Zhang et al. proposed a new proxy blind signature based on bilinear pairings, which satisfies distinguishability, verifiability, strong nonforgeability, strong identifiability, strong nondeniability, and prevention of misuse. Meanwhile, this scheme did not depend on public key infrastructure (PKI) [31]. Later, Li et al. introduced a new PBS, which was also constructed on bilinear pairings; it was independent of PKI [32]. However, these schemes are inefficient and are not proven to be secure.

IDPBS Schemes Based on Lattice. In 2014, Zhang and Ma initially proposed a proxy blind signature on lattice; it does not need to depend on PKI; its security is based on short integer solution problem. However, this scheme is still inefficient. [33].

3. Preliminaries

In the beginning, we will define the denotations that will be used all over the paper in Denotations.

3.1. NTRU Lattice and Rejection Sampling on Lattice

Definition 1 (NTRU lattice). The notations are defined as f, $g \in \mathbf{R}$ and $h = gf^{-1} \bmod q$; after that, the NTRU lattice can be defined as $\mathscr{L}_{h,q} = \{u, v \in \mathbf{R}^2 : u + vh = 0 \bmod q\}$. That is, $\mathscr{L}_{h,q}$ is on behalf of a $\mathbb{R}^{2N}$ full-rank lattice whose basis is $\left(\begin{smallmatrix} -\mathbf{T}_N(h) & \mathbf{I}_N \\ q\mathbf{I}_N & \mathbf{O}_N \end{smallmatrix}\right)$, $\mathbf{I}_N$ denotes a unit matrix, $\mathbf{O}_N$ denotes a null matrix, and $\mathbf{T}_N(h)$ denotes an anticirculant matrix $\begin{pmatrix} h_0 & h_1 & \cdots & h_{N-1} \\ -h_{N-1} & h_0 & \cdots & h_{N-2} \\ \vdots & \vdots & \vdots & \vdots \\ -h_1 & -h_2 & \cdots & h_0 \end{pmatrix}$.

Definition 2 (*R-SIS*$^{\kappa}_{q,1,2,\beta}$ on NTRU lattice). Small f and g can be sampled from $D_{\mathbb{Z}^N,\sigma}$ ($f, g \bmod q \in \mathbf{R}_q^{\times}$); then $\mathscr{L}_{h,q} = (h, 1) \in \mathbf{R}_q^{1\times 2}$ and $h = gf^{-1}$ can be obtained by using Algorithm 3 in [25]. Therefore, R-SIS on NTRU means finding $\mathbf{z_1}, \mathbf{z_2}$ satisfying $\mathscr{L}_{h,q}(\mathbf{z}_1, \mathbf{z}_2)^T = \mathbf{0} \bmod q$ and $\|(\mathbf{z_1}, \mathbf{z_2})\| \le \beta$.

Theorem 3 (rejection sampling theorem). *V denotes one subset of $\mathbb{Z}^m$, the norms of V's elements are less than constant T, $\sigma = \omega(T\sqrt{\log m}) \in \mathbb{R}$, ($M$ is invariable), and $h : V \longrightarrow \mathbb{R}$ is a probability distribution. Two algorithms are as follows: One is*

$$\begin{gathered} \mathbf{v} \longleftarrow h; \\ \mathbf{w} \longleftarrow D^N_{\mathbf{v},\sigma}; \\ \textit{get}\,(\mathbf{w}, \mathbf{v})\ \textit{with probability}\ \min\left(\frac{D^N_\sigma(\mathbf{w})}{MD^N_{\mathbf{v},\sigma}(\mathbf{w})}, 1\right). \end{gathered} \tag{1}$$

The other is

$$\begin{gathered} \mathbf{v} \longleftarrow h; \\ \mathbf{w} \longleftarrow D^N_\sigma; \\ \textit{get}\,(\mathbf{w}, \mathbf{v})\ \textit{with probability}\ \frac{1}{M}. \end{gathered} \tag{2}$$

Then the distribution of first algorithm will not exceed the second one's statistical distance $2^{-\omega(\log N)}/M$. Moreover, The first one will export something with probability at least $(1 - 2^{-\omega(\log N)})/M$.

3.2. The Definitions of IDPBS Scheme. An IDPBS consists of seven algorithms $(ST_\varepsilon, KE_\varepsilon, DG_\varepsilon, BS_\varepsilon, PS_\varepsilon, UB_\varepsilon, PV_\varepsilon)$

```
Setup: $ST_\varepsilon(1^N) \rightarrow (params, mk)$
Key   Extraction: $KE_\varepsilon(params, mk, id_i) \rightarrow sk_{id_i}, i = \mathcal{O}, \mathcal{P}$
Delegation   Generation :
                $DG_\varepsilon(id_\mathcal{O}, sk_{id_\mathcal{O}}, \omega) \rightarrow \psi$
                                        $\xrightarrow{\psi}$
                                                  proxy signer $\mathcal{P}$
Blind (user):
                $BS_\varepsilon(m, f) \rightarrow m'$
                                        $\xrightarrow{m'}$
Proxy Signature (proxy signer):
                                                  $PS_\varepsilon(m', id_P, sk_P, id_O, \psi) \rightarrow \sigma'$
                                        $\xleftarrow{\sigma'}$
Unblind (user):
                $UB(\sigma', f) \rightarrow \sigma$
Proxy Verify:
                                                  $PV_\varepsilon(id_\mathcal{O}, id_\mathcal{P}, \omega, m, \psi, \sigma) \rightarrow true/false$
```

Algorithm 1: General IDPBS scheme.

[12, 36, 37]. TTP will execute $ST_\varepsilon(1^N)$ to produce public parameters and keys [29, 38, 39]. The formal definition is presented as follows (Algorithm 1):

(i) $ST_\varepsilon(1^n)$ outputs $params$ and $mk = (msk, mpk)$.

(ii) $KE_\varepsilon(params, msk, id_i)$ outputs sk_{id_i} for $\mathcal{O}$ and $\mathcal{P}$ ($i = \mathcal{O}$ or $\mathcal{P}$).

(iii) $DG_\varepsilon(id_O, sk_{id_O}, w)$ outputs ψ for $\mathcal{P}$.

(iv) Proxy blind signature: $\mathcal{U}$ interacts with $\mathcal{P}$ according to the following protocol:

   (1) $BS(m, f)$: $\mathcal{U}$ blinds m to m' by using f and then sends m' to $\mathcal{P}$.

   (2) $PS_\varepsilon(m', id_P, sk_P, id_O, \psi)$: $\mathcal{P}$ signs on m' using sk_{id_P} and sends the signature σ' to $\mathcal{U}$.

   (3) $UB(\sigma', f)$: $\mathcal{U}$ unblinds σ' by using f and outputs the blind signature σ.

(v) $PV_\varepsilon(id_O, id_P, w, m, \psi, \sigma)$: if ψ, σ are valid, the algorithm outputs true. Otherwise it outputs false.

An IDPBS scheme should meet the following six properties. The details can be seen in [20, 33, 40–42].

(1) Blindness. P^* are denoted as an adversary who can control the proxy signer. P^* chooses two messages m_0 and m_1. Then a random bit $i \in \{0, 1\}$ is chosen in the game. m_0 and m_1 are randomly denoted as m_i and m_{1-i}. These two messages are, respectively, used as two user's inputs. After that, P^* will adaptively and parallelly interact with two honest users according to the signature protocol. Finally, two users output σ_i and σ_{1-i} respectively. Then σ_i and σ_{1-i} ordered by m_i and m_{1-i} are delivered to P^*; after that, P^* will output $i' \in \{0, 1\}$.

(2) One More Unforgeability. $\mathcal{P}$ can generate a legal proxy σ instead of $\mathcal{O}$. However, $\mathcal{O}$ and all the other entities fail to generate a legal signature. The game is presented as follows [33]: Adv_{U^*}, the advantage of U^*, is denoted as success probability in Algorithm 3. If no adversary can win Algorithm 3 at minimum with negligible probability η in time t, then it satisfies one more unforgeability [31].

(3) Verifiability. $\mathcal{V}$ can check whether σ is delegated by $\mathcal{O}$.

(4) Strong Identifiability. Any $\mathcal{V}$ can determine $\mathcal{P}$'s identity once he receives the proxy signature tuple.

(5) Strong Undeniability. $\mathcal{P}$ cannot refuse to admit it once he creates the proxy signature σ.

(6) Key Dependence. $\mathcal{P}$ can sign on a message if and only if he has the authorization from $\mathcal{O}$.

4. Proposed IDPBS-NTRU Scheme

Here, we introduce a novel IDPBS-NTRU $\varepsilon = (ST_\varepsilon, KE_\varepsilon, DG_\varepsilon, BS_\varepsilon, PS_\varepsilon, UB_\varepsilon, PV_\varepsilon)$, which can be seen in Algorithm 4. The details are as follows.

(1) $ST_\varepsilon(1^N)$. $q = \text{Poly}(N)$, $\varepsilon \in (0, \ln N / \ln q)$, and $s = \widetilde{\Omega}(N^{3/2}\sigma)$. If $N > 2, \sigma = N\sqrt{(\ln(8Nq)}q^{1/2+\varepsilon}, q^{1/2-\varepsilon} = \widetilde{\Omega}(n^{7/2})$. If $N = 2, \sigma = N\sqrt{\ln(8Nq)}q^{1/2+\varepsilon}, q^{1/2-\varepsilon} = \widetilde{\Omega}(N^3)$. $mk = (msk, mpk)$ can be obtained as below [25].

The algorithm takes samples f and g from $D_{\mathbb{Z}^N, s}$. Here, f, $g \bmod q \notin \mathbf{R}_q^\times$, $\|f\|, \|g\| \leq \sigma\sqrt{N}$, and $\langle f, g\rangle \in \mathbf{R}$. After that, the algorithm can get $F_1, G_1 \in \mathbf{R}$ according to the equation $fG_1 - gF_1 = 1$. Given $F_q = qF_1$ and $G_q = qG_1$, (F_q, G_q) can be obtained from $(f, g), (xf, xg), \ldots, (x^{N-1}f, x^{N-1}g)$ according to Babai algorithm [43]. Then there exists $(F, G) = (F_q, G_q) - k(f, g)$. If $\|(F, G)\| \leq N\sigma$, then the algorithm outputs system parameters $paras = (q, \varepsilon, s)$; the master private-key $\mathbf{msk}$ and master public-key mpk are as follows:

$$\mathbf{msk} = \mathbf{B} = \begin{pmatrix} \mathbf{T}(f) & \mathbf{T}(g) \\ \mathbf{T}(F) & \mathbf{T}(G) \end{pmatrix},$$
$$mpk = h = gf^{-1} \in \mathbf{R}_q^{\times}. \tag{3}$$

(2) $KE_{\varepsilon}(paras, id_i, \boldsymbol{msk})$. The algorithm executes (4) to get an N-dimension matrix $\mathbf{t}$; then the algorithm executes (6) and outputs sk_{id_i} according to corresponding id_i $(i = \mathcal{O}, \mathcal{P})$ [25].

$$\mathbf{t} \longleftarrow H_1(id_i), \tag{4}$$

$$sk_{id_i} = \left(s_{i_1}, s_{i_2}\right) \longleftarrow [(t, 0) - \text{Gausssian}(\mathbf{msk}, \sigma, (t, 0))],$$
$$s_{i_1} + s_{i_2} h = t. \tag{5}$$

(3) DG_{ε}. ω is denoted as a warrant. $\mathcal{O}$ will execute this algorithm to generate a valid delegation.

(i) The algorithm chooses $\mathbf{y}_1, \mathbf{y}_2 \in D_{\mathbb{Z}^N,s}$ at random.

(ii) The algorithm executes (6) to get an N-dimension matrix $\mathbf{u}$.

(iii) The algorithm executes (7) and (8) to generate valid delegations $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$. Here, the algorithm uses the rejection sampling theorem to keep the delegation independent on $\mathcal{O}$'s secret keys $\mathbf{s}_{\mathcal{O}_1}$ and $\mathbf{s}_{\mathcal{O}_2}$.

(iv) $\mathcal{O}$ sends $(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, \mathbf{u}, \omega)$ to $\mathcal{P}$.

$$\mathbf{u} = H_2(\mathbf{y}_1 + h\mathbf{y}_2, \omega), \tag{6}$$

$$\boldsymbol{\sigma}_1 = \mathbf{y}_1 + \mathbf{s}_{\mathcal{O}_1}\mathbf{u}, \tag{7}$$

$$\boldsymbol{\sigma}_2 = \mathbf{y}_2 + \mathbf{s}_{\mathcal{O}_2}\mathbf{u}. \tag{8}$$

(4) BS_{ε}. m is a plaintext. $\mathcal{U}$ will execute this algorithm to generate a blind message, which needs to be signed by $\mathcal{P}$.

(i) The algorithm will randomly select $\mathbf{y}_3, \mathbf{y}_4, \boldsymbol{\alpha}, \boldsymbol{\gamma} \in D_{\mathbb{Z}^N,s}$.

(ii) The algorithm executes (9) to get an N-dimension $\mathbf{e}$.

(iii) The algorithm executes (10) to blind $\mathbf{e}$.

(iv) $\mathcal{U}$ sends $(\mathbf{y}_3, \mathbf{y}_4, \mathbf{e}^*)$ to a proxy signer $\mathcal{P}$.

$$\mathbf{e} = H_3(\mathbf{y}_3 + h\mathbf{y}_4 + h\boldsymbol{\gamma} + \boldsymbol{\alpha} - \boldsymbol{\alpha}H(id), m), \tag{9}$$

$$\mathbf{e}^* = \mathbf{e} - \boldsymbol{\alpha}. \tag{10}$$

(5) PS_{ε}. The proxy signer $\mathcal{P}$ will execute this algorithm to sign on the blinded message.

(i) The algorithm validates whether (11) and (12) are true. If either is false, $\mathcal{P}$ aborts the algorithm. Otherwise, it continues.

(ii) The algorithm will execute (13) and (14). Here, the rejection sample theory is used to keep the proxy signatures $\boldsymbol{\sigma}_3$ and $\boldsymbol{\sigma}_4$ independent on $\mathcal{P}$'s secret keys $\mathbf{s}_{\mathcal{P}_1}$ and $\mathbf{s}_{\mathcal{P}_2}$.

(iii) The algorithm outputs the tuple $(m, \omega, \boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, \mathbf{u}, \boldsymbol{\sigma}_3, \boldsymbol{\sigma}_4, \mathbf{e})$.

$$\left\|(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)\right\| \le 2s\sqrt{2N}, \tag{11}$$

$$H_2(h\boldsymbol{\sigma}_2 + \boldsymbol{\sigma}_1 - H_1(id_{\mathcal{O}})\mathbf{u}, \omega) = \mathbf{u}, \tag{12}$$

$$\boldsymbol{\sigma}_3 = \mathbf{y}_3 + s_{\mathcal{P}_1}\mathbf{e}, \tag{13}$$

$$\boldsymbol{\sigma}_4 = \mathbf{y}_4 + s_{\mathcal{P}_2}\mathbf{e}. \tag{14}$$

(6) UB_{ε}. $\mathcal{U}$ will execute the algorithm to unblind the proxy signature.

(i) The algorithm executes (15) to unblind the proxy signature tuple.

(ii) $\mathcal{U}$ outputs the signature tuple $(\boldsymbol{\sigma}_1, \mathbf{m}, \boldsymbol{\sigma}_2, \boldsymbol{\omega}, \mathbf{u}, \boldsymbol{\sigma}_3, \boldsymbol{\sigma}_4, \mathbf{e})$.

$$\begin{aligned} \boldsymbol{\sigma}_3 &= \boldsymbol{\sigma}_3^* + \boldsymbol{\alpha}, \\ \boldsymbol{\sigma}_4 &= \boldsymbol{\sigma}_4^* + \boldsymbol{\gamma}. \end{aligned} \tag{15}$$

(7) $PV_{\varepsilon}(\boldsymbol{\sigma}_1, m, \boldsymbol{\sigma}_2, \omega, \mathbf{u}, \boldsymbol{\sigma}_3, \boldsymbol{\sigma}_4, \mathbf{e}, id_{\mathcal{O}}, id_{\mathcal{P}})$. $\mathcal{V}$ will execute this algorithm to validate whether the signature tuple satisfies (16). If all the equations mentioned above are true, the signatures are valid. Otherwise, they are invalid.

$$\begin{aligned} &\left\|(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)\right\| \le 2s\sqrt{2N}, \\ &H_2(h\boldsymbol{\sigma}_2 + \boldsymbol{\sigma}_1 - H_1(id_{\mathcal{O}})u, w) = \mathbf{u}, \\ &\left\|(\boldsymbol{\sigma}_3, \boldsymbol{\sigma}_4)\right\| \le 2s\sqrt{2N}, \\ &H_3(h\boldsymbol{\sigma}_4 + \boldsymbol{\sigma}_3 - H_1(id_{\mathcal{P}})\mathbf{e}, m) = \mathbf{e}. \end{aligned} \tag{16}$$

5. Security and Performance Comparison

5.1. Security

(1)

Theorem 4 (correctness). *The IDPBS-NTRU scheme is correct.*

Proof. According to the construction of our IDPBS scheme, we can get

$$\begin{aligned} &h * \boldsymbol{\sigma}_2 + \boldsymbol{\sigma}_1 - H(id_{\mathcal{O}}) * \mathbf{u} \\ &\quad = h\left(\mathbf{y}_2 + \mathbf{s}_{\mathcal{O}_2}\mathbf{u}\right) + \mathbf{y}_1 + \mathbf{s}_{\mathcal{O}_1}\mathbf{u} - H(id_{\mathcal{O}}) * \mathbf{u} \\ &\quad = \mathbf{y}_1 + h\mathbf{y}_2. \end{aligned} \tag{17}$$

Therefore, $H_2(h * \boldsymbol{\sigma}_2 + \boldsymbol{\sigma}_1 - H(id_{\mathcal{O}}) * \mathbf{u}, m) = \mathbf{u}$.

$$\begin{aligned} &h\boldsymbol{\sigma}_4 + \boldsymbol{\sigma}_3 - H_1(id_{\mathcal{P}})\mathbf{e} \\ &\quad = h(\boldsymbol{\sigma}_4^* + \boldsymbol{\gamma}) + \boldsymbol{\sigma}_3^* + \boldsymbol{\alpha} - H(id_{\mathcal{P}}) * \mathbf{e} \\ &\quad = h\left(\mathbf{y}_4 + s_{\mathcal{P}_2}\mathbf{e}^* + \boldsymbol{\gamma}\right) + \mathbf{y}_3 + \boldsymbol{\alpha} + s_{\mathcal{P}_1}\mathbf{e}^* - H(id_{\mathcal{P}}) \\ &\qquad * \mathbf{e} = \mathbf{y}_3 + h\mathbf{y}_4 + h\boldsymbol{\gamma} + \boldsymbol{\alpha} - \boldsymbol{\alpha}H_1(id_{\mathcal{P}}). \end{aligned} \tag{18}$$

Thus, $H_3(h\boldsymbol{\sigma}_4 + \boldsymbol{\sigma}_3 - H_1(id_{\mathcal{P}})\mathbf{e}, m) = \mathbf{e}$. □

```
$i \in_{\$} \{0, 1\}$
$(params, msk) \leftarrow ST(1^n)$
$sk_{id_b} \leftarrow EX(params, id_b, msk), b = \mathcal{O}$ or $\mathcal{P}$
$\psi \leftarrow DG_{\varepsilon}(id_{\mathcal{O}}, sk_{id_{\mathcal{O}}}, \omega)$
$(m_0, m_1, \omega, \psi, state_{find}) \leftarrow_{\$} \mathcal{P}^{*}(find, sk_{id_{\mathcal{P}}}, id_{\mathcal{P}}, id_{\mathcal{O}}, \psi)$
$state_{issue} \leftarrow_{\$} \mathcal{P}^{*<.,\mathcal{U}(id_{\mathcal{P}},m_i)^1>,<.,\mathcal{U}(id_{\mathcal{P}},m_{1-i})^1>}(issue, state_{find})$
$\sigma_i, \sigma_{1-i}$ are respectively $\mathcal{U}(id_{\mathcal{P}}, m_i), \mathcal{U}(id_{\mathcal{P}}, m_{1-i})$'s output
if $\sigma_0 \neq$ fail and $\sigma_1 \neq fail$ then
    $i' \leftarrow_{\$} \mathcal{P}^{*}(guess, \sigma_0, \sigma_1, state_{issue})$
else
    $i' \leftarrow_{\$} \mathcal{P}^{*}(guess, fail, fail, state_{issue})$
end if
return true iff $i' = i$
```

Algorithm 2: $\mathrm{Expt}_{\mathcal{S}^*}^{bd}(n)$.

```
$(params, msk) \leftarrow ST(1^n)$
$sk_{id_b} \leftarrow EX(params, id_b, msk), b = \mathcal{O}$ or $\mathcal{P}$
$\psi \leftarrow DG_{\varepsilon}(id_{\mathcal{O}}, sk_{id_{\mathcal{O}}}, \omega)$
$\{(m_1, \sigma_1), ..., (m_k, \sigma_k)\} \leftarrow_{\$} \mathcal{U}^{*h(.),<\mathcal{S}(sk_{id_{\mathcal{P}}}),.>^{\infty}}(id_{\mathcal{P}})$
$l$ is the successful interaction number between $\mathcal{U}^*$ and $\mathcal{P}$
return true iff
    $m_i \neq m_j$ for $1 \leq i < j \leq k$ and
    $VF(m_i, \sigma_i, id) = 1$ and
    $l + 1 = k$
```

Algorithm 3: $\mathrm{Expt}_{\mathcal{U}^*}^{omf}(n)$.

(2)

Theorem 5 (blindness). *The IDPBS-NTRU scheme satisfies blindness.*

Proof. As shown in Algorithm 2, A random bit $i \in \{0,1\}$ which is kept secret from P^*. Then P^* chooses two messages m_0 and m_1. m_0 and m_1 are randomly denoted as m_i and m_{1-i}. m_i and m_{1-i} are the inputs of two honest users, respectively. P^* adaptively and parallelly interacts with two honest users, respectively. Finally, these two honest users output σ_i and σ_{1-i}, respectively. The sequence σ_i and σ_{1-i} ordered by m_i and m_{1-i} will be sent to P^*. P^* will output a bit $i' \in \{0,1\}$.

In the process of signature protocol, all intermediate results do not depend on m; thus it is enough to analyze $\mathbf{e}^*$, $\mathbf{y}_3$, $\mathbf{y}_4$, $\boldsymbol{\sigma}_1$, $\boldsymbol{\sigma}_2$, $\mathbf{u}$, $\boldsymbol{\omega}$, $\boldsymbol{\sigma}_3^*$, and $\boldsymbol{\sigma}_4^*$.

To $\mathbf{e}^*$, the statistical distance is presented as follows:

$$\Delta\left(\mathbf{e}_i^*, \mathbf{e}_{1-i}^*\right) = \frac{1}{2} \sum_{\mathbf{e}^{*'} \in D_{\mathbb{Z}^N,s}} \left|\Pr\left(\mathbf{e}_i^* = \mathbf{e}^{*'}\right) - \Pr\left(\mathbf{e}_{1-i}^* = \mathbf{e}^{*'}\right)\right|. \tag{19}$$

□

Since α is chosen at random, next we obtain the equations $\Pr(\mathbf{e}_i^* = \mathbf{e}^{*'}) = 1/2$ and $\Pr(\mathbf{e}_{1-i}^* = \mathbf{e}^{*'}) = 1/2$. Therefore, we obtain $\Delta(\mathbf{e}_i^*, \mathbf{e}_{1-i}^*) = 0$.

In the same way, we can obtain $\Delta(\mathbf{y}_3^i, \mathbf{y}_3^{1-i}) = 0$, $\Delta(\mathbf{y}_4^i, \mathbf{y}_4^{1-i}) = 0$, $\Delta(\boldsymbol{\sigma}_1^{i*}, \boldsymbol{\sigma}_1^{1-i*}) = 0$, $\Delta(\boldsymbol{\sigma}_2^{i*}, \boldsymbol{\sigma}_2^{1-i*}) = 0$, $\Delta(\mathbf{u}^{i*}, \mathbf{u}^{1-i*}) = 0$, $\Delta(\boldsymbol{\omega}^{i*}, \boldsymbol{\omega}^{1-i*}) = 0$, $\Delta(\boldsymbol{\sigma}_3^{i*}, \boldsymbol{\sigma}_3^{1-i*}) = 0$, and $\Delta(\boldsymbol{\sigma}_4^{i*}, \boldsymbol{\sigma}_4^{1-i*}) = 0$. Therefore, $\mathcal{S}^*$ cannot distinguish m among $\mathbf{e}^*$, $\mathbf{y}_3$, $\mathbf{y}_4$, $\boldsymbol{\sigma}_1$, $\boldsymbol{\sigma}_2$, $\mathbf{u}$, $\boldsymbol{\omega}$, $\boldsymbol{\sigma}_3^*$, and $\boldsymbol{\sigma}_4^*$; that is, $\mathcal{S}^*$ can win this experiment with probability 1/2. Thus, the theorem is proven.

We denote δ_1, δ_2, δ_3, δ_4, δ_5, and δ_6 as the cost functions of H_1, H_2, H_3 hash oracle, extract Oracle, DG oracle, and signature oracle, respectively, η as a nonnegligible probability, Θ as a polynomial time algorithm, and Γ as a polynomial time forger.

(3)

Theorem 6 (one more unforgeability). *If Γ is able to generate a legal IDPBS signature with η in t successfully, after at most $\tau_1, \tau_2, \tau_3, \tau_4, \tau_5, \tau_6$ times queries, respectively, to H_1, H_2, H_3 hash, extract, DG, and blind signature oracles, then R-SIS$_{q,1,2,\beta}^{\kappa}$ can be solved by Θ with probability at least $\eta' = ((1 - 2^{-\omega(\log N)})\eta)^2$ in time $t' = t+\tau_1^{\tau_4+\tau_6}(\tau_1\xi_1+\tau_6\xi_6+\tau_4\xi_4)+\tau_2^{\tau_5}(\tau_2\xi_2+\tau_5\xi_5)$.*

Proof. We suppose that Γ is able to generate an IDPBS signature successfully with η; we are able to construct Θ to calculate *R-SIS*'s solution. The interaction environment can be simulated as follows.

Setup. Θ selects $h \in \mathbf{R}_q^{\times}$ and H_1, H_2, and H_3 at random. Next Θ calculates and delivers $paras = \{h, H_1, H_2, \epsilon, q, s\}$ to Γ.

Queries on H_1 Oracle. To reply to H_1 oracle's query, Θ creates one null list L_1. Once Θ obtains one id_i, Θ will query H_1. If there is a t_i consistent with the query, Θ will return t_i. Otherwise, Θ will select a random t_i. At last, Θ will return t_i to Γ and save (id_i, t_i) to L_1.

Queries on H_2 Oracle. To reply to H_2's queries, Θ creates a list L_2; is initialized null. When Θ receives an $(\mathbf{y}_{i_1}, \mathbf{y}_{i_2}, w_i)$, Θ will query H_2. If there is one corresponding value t_i, Θ will return $\mathbf{u}_i$. Else Θ will choose one $\mathbf{u}_i$ at random. Finally, Θ will return $\mathbf{u}_i$ to Γ and save $(\mathbf{y}_{i_1}, \mathbf{y}_{i_2}, w_i, \mathbf{u}_i)$ to L_2.

KE Oracle Queries. When Θ queries a private key related to one id_i, Γ will recover the corresponding (id_i, t_i) from L_1. Next Γ will run $sk_{id_i} = (s_{i_1}, s_{i_2}) \leftarrow [(t, 0) - Gausssian(sk, \sigma, (t, 0))]$; Γ will return id_i, t_i, sk_{id_i} to Θ and save the tuple to L_3.

DG Oracle Queries. When Θ requests the delegation queries, Θ will verify whether id_i has been queried for H_1 or KE_{ε} oracle. If it has been queried, Θ will obtain (id_i, t_i, sk_{id_i}) from L_3. Else Θ will simulate KE_{ε} oracle and get a new private key. Next Θ will execute $\boldsymbol{\sigma}_{i_1} = \mathbf{y}_{i_1} + s_{\mathcal{O}_{i_1}} \mathbf{u}_i$ and $\boldsymbol{\sigma}_{i_2} = \mathbf{y}_{i_2} + s_{\mathcal{O}_{i_2}} \mathbf{u}_i$ to get a legal delegation signature $(w_i, \mathbf{u}_i, \boldsymbol{\sigma}_{i_1}, \boldsymbol{\sigma}_{i_2})$ and save $(\mathbf{y}_{i_1}, \mathbf{y}_{i_2}, m_i, \boldsymbol{\sigma}_{i_1}, \boldsymbol{\sigma}_{i_2})$ to L_4.

```
$Setup: ST_{\varepsilon}(1^N) \rightarrow (paras, mk)$
$Key\ Extraction: t \leftarrow H_1(id_i), sk_{id_i} = (s_{i_1}, s_{i_2}) \leftarrow [(t,0) - Gausssian(\mathbf{msk}, \sigma, (t,0))]$
$Delegation\ Generation\,(original\ signer O):$
    $\mathbf{y}_1, \mathbf{y}_2 \in \mathrm{D}_{\mathbb{Z}^N,s}$
    $\mathbf{u} = H_2(\mathbf{y}_1 + h\mathbf{y}_2, \omega)$
    $\boldsymbol{\sigma}_1 = \mathbf{y}_1 + s_{\mathscr{O}_1}\mathbf{u}$
    $\boldsymbol{\sigma}_2 = \mathbf{y}_2 + s_{\mathscr{O}_2}\mathbf{u}$
                                        $\xrightarrow{\omega,\mathbf{u},\boldsymbol{\sigma}_1,\boldsymbol{\sigma}_2}$
                                                        proxy signer $\mathscr{P}$
Blind (user $\mathscr{U}$):
    $\mathbf{y}_3, \mathbf{y}_4, \boldsymbol{\alpha}, \boldsymbol{\gamma} \in D_{\mathbb{Z}^N,s}$
    $\mathbf{e} = H_3(\mathbf{y}_3 + h\mathbf{y}_4 + h\boldsymbol{\gamma} + \boldsymbol{\alpha} - \boldsymbol{\alpha}H(id), m)$
    $\mathbf{e}^* = \mathbf{e} - \boldsymbol{\alpha}$
                                        $\xrightarrow{\mathbf{e}^*,\mathbf{y}_3,\mathbf{y}_4}$
Proxy Signature (proxy signer $\mathscr{P}$):
                                                        $\boldsymbol{\sigma}_3^* = \mathbf{y}_3 + s_{\mathscr{P}_1}\mathbf{e}^*$
                                                        $\boldsymbol{\sigma}_4^* = \mathbf{y}_4 + s_{\mathscr{P}_2}\mathbf{e}^*$
                                        $\xleftarrow{\boldsymbol{\sigma}_1,\boldsymbol{\sigma}_2,\mathbf{u},\omega,\boldsymbol{\sigma}_3^*,\boldsymbol{\sigma}_4^*}$
Unblind (user $\mathscr{U}$):
    $\boldsymbol{\sigma}_3 = \boldsymbol{\sigma}_3^* + \boldsymbol{\alpha}$
    $\boldsymbol{\sigma}_4 = \boldsymbol{\sigma}_4^* + \boldsymbol{\gamma}$
    $signature \quad tuple(\omega, m, \boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, \mathbf{u}, \boldsymbol{\sigma}_3, \boldsymbol{\sigma}_4, \mathbf{e})$
$Proxy\ Verify\,(verifier\ \mathscr{V}):$
                                                        if $\|(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)\| \le 2s\sqrt{2N}$
                                                        & $\|(\boldsymbol{\sigma}_3, \boldsymbol{\sigma}_4)\| \le 2s\sqrt{2N}$
                                                        & $H_2(h\boldsymbol{\sigma}_2 + \boldsymbol{\sigma}_1 - H_1(id_{\mathscr{O}})\mathbf{u}, \omega) = \mathbf{u}$
                                                        & $H_3(h\boldsymbol{\sigma}_4 + \boldsymbol{\sigma}_3 - H_1(id_{\mathscr{P}})\mathbf{e}, m) = \mathbf{e}$
                                                        output true, otherwise false
```

Algorithm 4: IDPBS-NTRU protocol.

Signature Oracle Queries. Γ queries the signing oracle for $\mathbf{y}_{i_3}$, $\mathbf{y}_{i_4}$, α_i, β_i, id_i, and m_i. Θ will verify whether id_i has been queried for H_1 or KE_{ε} oracle. If it has been queried, Θ will obtain (id_i, t_i, sk_{id_i}) from L_3. Else Θ will simulate the EX_{ϵ} oracle and get a new private key. Next Θ will execute DG queries and then obtain $(\mathbf{y}_{i_1}, \mathbf{y}_{i_2}, \omega_i, u_i, \boldsymbol{\sigma}_{i_1}, \boldsymbol{\sigma}_{i_2})$ from L_4. Then Θ will execute $\boldsymbol{\sigma}_{i_3} = \mathbf{y}_{i_3} + s_{\mathscr{P}_{i_1}}\mathbf{e}_i$ and $\boldsymbol{\sigma}_{i_4} = \mathbf{y}_{i_4} + s_{\mathscr{P}_{i_2}}\mathbf{e}_i$ to get a valid signature $(m_i, \mathbf{u}_i, \boldsymbol{\sigma}_{i_3}, \boldsymbol{\sigma}_{i_4}, \mathbf{e}_i)$ and save $(\mathbf{y}_{i_1}, \mathbf{y}_{i_2}, \omega_i, u_i, \boldsymbol{\sigma}_{i_1}, \boldsymbol{\sigma}_{i_2}, \mathbf{y}_{i_3}, \boldsymbol{\alpha}_i, \boldsymbol{\beta}_i, \mathbf{y}_{i_4}, m_i, \boldsymbol{\sigma}_{i_3}, \boldsymbol{\sigma}_{i_4}, \mathbf{e}_i)$ to L_5.

Output. At last, Γ will output one forged signature $(\mathbf{u}_i, \omega_i, \boldsymbol{\sigma}_{i_1}, \boldsymbol{\sigma}_{i_2}, m_i, \mathbf{e}_i, \boldsymbol{\sigma}_{i_3}, \boldsymbol{\sigma}_{i_4})$ on w_i, m_i, $id_{\mathscr{O}_i}$, and $id_{\mathscr{P}_i}$ for the first time. Θ will rewind Γ to the point where w_i and m_i are queried for H_1; next Γ will get a valid tuple $(\mathbf{u}_i', \omega_i', \boldsymbol{\sigma}_{i_1}', \boldsymbol{\sigma}_{i_2}', m_i', \mathbf{e}_i', \boldsymbol{\sigma}_{i_3}', \boldsymbol{\sigma}_{i_4}')$ once again.

Thus, Θ are able to solve $R\text{-}SIS^{\kappa}_{q,1,2,\beta}$ problem. Θ will compute $\boldsymbol{\sigma}_{i_3^*} = \mathbf{y}_{i_3} + s_{O_{i_1}}\mathbf{e}_i$, $\boldsymbol{\sigma}_{i_4} = \mathbf{y}_{i_4} + s_{P_{i_2}}\mathbf{e}_i$, and $\boldsymbol{\sigma}_{i_3} + \boldsymbol{\sigma}_{i_4}h - H(id_{\mathscr{P}_i})\mathbf{e}_i$. Next Θ will verify whether $\boldsymbol{\sigma}_{i_3} + \boldsymbol{\sigma}_{i_4}h - H(id_{\mathscr{P}_i})\mathbf{e}_i = \boldsymbol{\sigma}_{i_3}' + \boldsymbol{\sigma}_{i_4}'h - H(id_{\mathscr{P}_i})\mathbf{e}_i = \mathbf{y}_{i_3} + h\mathbf{y}_{i_4} + h\gamma_i + \alpha_i - \alpha_i H(id_{P_i})$. If $(\boldsymbol{\sigma}_{i_3}, \boldsymbol{\sigma}_{i_4}) \neq (\boldsymbol{\sigma}_{i_3}', \boldsymbol{\sigma}_{i_4}')$, we can obtain $\|(\boldsymbol{\sigma}_{i_3} - \boldsymbol{\sigma}_{i_3}', \boldsymbol{\sigma}_{i_4} - \boldsymbol{\sigma}_{i_4}')\| \le 4s\sqrt{2N}$. After that, $(\boldsymbol{\sigma}_{i_4} - \boldsymbol{\sigma}_{i_4}', \boldsymbol{\sigma}_{i_3} - \boldsymbol{\sigma}_{i_3}')$ is a solution to $R\text{-}SIS^{\kappa}_{q,1,2,\beta}$. Similarly, we can obtain that $(\boldsymbol{\sigma}_{i_2} - \boldsymbol{\sigma}_{i_2}', \boldsymbol{\sigma}_{i_1} - \boldsymbol{\sigma}_{i_1}')$ is a solution to $R\text{-}SIS^{\kappa}_{q,1,2,\beta}$.

After that, we begin to analyze Θ's advantages. As mentioned above, Θ will win this game if Γ has already forged a valid $(\boldsymbol{\sigma}_{i_1}', \omega_i, m_i, \boldsymbol{\sigma}_{i_2}', \mathbf{u}_i', \boldsymbol{\sigma}_{i_3}', \boldsymbol{\sigma}_{i_4}', \mathbf{e}_i')$ and $(\boldsymbol{\sigma}_{i_1}, \boldsymbol{\sigma}_{i_2}) \neq (\boldsymbol{\sigma}_{i_1}', \boldsymbol{\sigma}_{i_2}')$ and $(\boldsymbol{\sigma}_{i_3}, z_{i_4}) \neq (\boldsymbol{\sigma}_{i_3}', \boldsymbol{\sigma}_{i_4}')$. The simulation of the EX_{ϵ} oracle fails if H_2 causes inconformity. Then Θ is able to solve $R\text{-}SIS^{\kappa}_{q,1,2,\beta}$ with probability at minimum $\eta' = ((1 - 2^{-\omega(\log N)})\eta)^2$ [25]; here, $\beta = 4s\sqrt{2N}$; it is clear that $t' = t + \tau_1^{\tau_4+\tau_6}(\tau_1\xi_1 + \tau_6\xi_6 + \tau_4\xi_4) + \tau_2^{\tau_5}(\tau_2\xi_2 + \tau_5\xi_5)$. Therefore, we can prove the theorem.

(4) Verifiability. Once receiving $(w, m, \boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, \mathbf{u}, \boldsymbol{\sigma}_3, \boldsymbol{\sigma}_4, \mathbf{e}, id_{\mathscr{O}}, id_{\mathscr{P}})$, V will execute PV_{ε} to check whether $\|(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2)\| \le 2s\sqrt{2N}$ and $H_2(h\boldsymbol{\sigma}_2 + \boldsymbol{\sigma}_1 - H_1(id_{\mathscr{O}})\mathbf{u}, w) = \mathbf{u}$ are true. If both are true, the proxy signer is delegated by $\mathscr{O}$ to sign on m.

(5) Strong Identifiability. After receiving $(w, m, \boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, u, \boldsymbol{\sigma}_3, \boldsymbol{\sigma}_4, \mathbf{e}, id_{\mathscr{O}}, id_{\mathscr{P}})$, V can confirm $\mathscr{P}$'s identity in accordance with $id_{\mathscr{P}}$; thus the IDPBS-NTRU scheme satisfies strong identifiability.

(6) Strong Undeniability. $\boldsymbol{\sigma}_3$ and $\boldsymbol{\sigma}_4$ are signed by using $\mathscr{P}$'s secret keys $s_{\mathscr{P}_1}$ and $s_{\mathscr{P}_2}$; they will only be known by $\mathscr{P}$; thus $\mathscr{P}$ cannot refuse his signature once he signed; thus the IDPBS-NTRU scheme satisfies strong undeniability.

Table 1: Performance comparison with other lattice-based IDPBS schemes.

	ZM [33]	TA [34]	MMHP [35]	Ours
Problem	SIS	DLP	DLP	R-SIS
OSS	$m^2 \log(\lambda + 1)$	$3m$	m	$2N \log(12\sigma) + N(\log \lambda + 1)$
OSK	$m^2 \log(\lambda + 1)$	m	m	$2N \log(s\sqrt{N})$
PSS	$2m \log(\lambda + 1)$	m	m	$4N \log(12\sigma) + 2N(\log \lambda + 1)$
PSK	$m^2 \log(\lambda + 1)$	$7m$	m	$2N \log\left(s\sqrt{N}\right)$

(7) Key Dependence. $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$ on warrant ω are signed by $\mathcal{O}$'s secret keys $s_{\mathcal{O}_1}$ and $s_{\mathcal{O}_2}$; they are only known by $\mathcal{O}$,; $\mathcal{P}$ has no legal right to sign on a message before he is authorized by $\mathcal{O}$; thus the IDPBS-NTRU scheme satisfies key dependence. □

5.2. Performance. In this section, we compare the performance of IDPBS-NTRU with other IDPBS schemes. λ is written as security parameter, we denote $\mathcal{O}$'s signature size and signing-key size as OSS and OSK, respectively. $\mathcal{P}$'s signature size and signing key size are denoted as PSS and PSK, respectively. In ZM scheme [33], the parameters satisfy $m \geq 2N \lg q$ and $q > \beta\omega \log n$. In TA [34] and MMHP [35] schemes, the security parameter m is equal to N.

In Table 1, we compute the signature size and signing key size for $\mathcal{O}$ and $\mathcal{P}$. It is clear to draw a conclusion that our proxy signer's OSS, OSK, and PSK are smaller than ZM, TA, and MMHP schemes, our PSS is larger than ZM scheme, our PSS is smaller than TA scheme and MMHP scheme, and our OSS, OSK, PSK, and PSS are larger than TA and MMHP schemes. However, TA scheme and MMHP scheme are based on DLP; they are considered as not secure to resist the quantum computer attack. So our scheme can be more secure than them.

6. Conclusions

In this work, we present an IDPBS-NTRU scheme by using NTRU lattice; this scheme plays an important role in offline e-payment system, which can be deployed in fog computing model. We demonstrate that IDPBS-NTRU is efficient and secure. In addition, our IDPBS-NTRU's OSS, OSK, and PSK are smaller than ZM scheme and safer than TA and MMHP schemes. The proposed scheme is constructed based on NTRU lattice, which has the advantages of NTRU lattice. In the future, we will continue to construct a partial IDPBS scheme based on lattice.

Denotations

$\mathcal{O}$:	Original signer
$\mathcal{P}$:	Proxy signer
$\mathcal{U}$:	A user
$\mathcal{V}$:	A certifier
TTP:	Trusted third party
$params$:	System parameters
mk:	Master key
mpk:	Master public-key
msk:	Master secret-key
id:	User's identity
sk_{id}:	Secret key related to a user
ω:	Warrant
m:	A message
m':	A blinded message
f:	A blind factor
ψ:	Delegation
σ:	Blind signature
$R = Z[x]/(x^N + 1)$:	A ring
$f = \sum_{i=0}^{N-1} f_i x^i$:	A polynomial in R
$g = \sum_{i=0}^{N-1} g_i x^i$:	A polynomial in R
$\widetilde{\Omega}(\cdot)$:	The asymptotic lower bound
$Poly(N)$:	A polynomial function related to N
A:	An adversary
C:	A challenger
c:	A constant
N:	Security parameter.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (no. U1636213).

References

[1] M. Z. A. Bhuiyan, G. Wang, J. Wu, J. Cao, X. Liu, and T. Wang, "Dependable Structural Health Monitoring Using Wireless Sensor Networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 14, no. 4, pp. 363–376, 2017.

[2] X. Liu, K. R. Choo, R. H. Deng, R. Lu, and J. Weng, "Efficient and Privacy-Preserving Outsourced Calculation of Rational Numbers," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 1, pp. 27–39, 2018.

[3] X. Liu, R. H. Deng, K.-K. R. Choo, and J. Weng, "An efficient privacy-preserving outsourced calculation toolkit with multiple keys," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 11, pp. 2401–2414, 2016.

[4] Y. Zhang and Z. Han, "Multi-dimensional Payment Plan in Fog Computing with Moral Hazard," in *Contract Theory for Wireless Networks*, Wireless Networks, pp. 73–88, Springer International Publishing, Cham, 2017.

[5] Y. Xue, Y.-a. Tan, C. Liang, Y. Li, J. Zheng, and Q. Zhang, "RootAgency: A Digital signature-based root privilege management agency for cloud terminal devices," *Information Sciences*, vol. 444, pp. 36–50, 2018.

[6] S. Park and Y. Yoo, “Network Intelligence Based on Network State Information for Connected Vehicles Utilizing Fog Computing,” *Mobile Information Systems*, vol. 2017, Article ID 7479267, 9 pages, 2017.

[7] Y. Tan, Y. Xue, C. Liang et al., “A root privilege management scheme with revocable authorization for Android devices,” *Journal of Network and Computer Applications*, vol. 107, pp. 69–82, 2018.

[8] J. Li, L. Sun, Q. Yan, Z. Li, W. Srisa-an, and H. Ye, “Significant Permission Identification for Machine Learning Based Android Malware Detection,” *IEEE Transactions on Industrial Informatics*, vol. PP, no. 99, pp. 1-1, 2018.

[9] M. Z. Alam Bhuiyan, J. Wu, G. Wang, and J. Cao, “Sensing and Decision Making in Cyber-Physical Systems: The Case of Structural Event Monitoring,” *IEEE Transactions on Industrial Informatics*, vol. 12, no. 6, pp. 2103–2114, 2016.

[10] X. Chen, X. Huang, J. Li, J. Ma, W. Lou, and D. S. Wong, “New algorithms for secure outsourcing of large-scale systems of linear equations,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 1, pp. 69–78, 2014.

[11] Z. Guan, J. Li, L. Zhu, Z. Zhang, X. Du, and M. Guizani, “Toward Delay-Tolerant Flexible Data Access Control for Smart Grid With Renewable Energy Resources,” *IEEE Transactions on Industrial Informatics*, vol. 13, no. 6, pp. 3216–3225, 2017.

[12] R. Zhu, B. Zhang, J. Mao, Q. Zhang, and Y.-A. Tan, “A methodology for determining the image base of ARM-based industrial control system firmware,” *International Journal of Critical Infrastructure Protection*, vol. 16, pp. 26–35, 2017.

[13] T. Li, J. Li, Z. Liu, P. Li, and C. Jia, “Differentially private Naive Bayes learning over multiple data sources,” *Information Sciences*, vol. 444, pp. 89–104, 2018.

[14] J. Xu, L. Wei, Y. Zhang, A. Wang, F. Zhou, and C. Gao, “Dynamic Fully Homomorphic encryption-based Merkle Tree for lightweight streaming authenticated data structures,” *Journal of Network and Computer Applications*, vol. 107, pp. 113–124, 2018.

[15] X. Yu, Y. Tan, C. Zhang et al., “A High-Performance Hierarchical Snapshot Scheme for Hybrid Storage Systems,” *Journal of Electronics*, vol. 27, no. 1, pp. 76–85, 2018.

[16] Q. Lin, J. Li, Z. Huang, W. Chen, and J. Shen, “A short linearly homomorphic proxy signature scheme,” *IEEE Access*, vol. 6, pp. 12966–12972, 2018.

[17] Z. Guan, J. Li, L. Wu, Y. Zhang, J. Wu, and X. Du, “Achieving Efficient and Secure Data Acquisition for Cloud-Supported Internet of Things in Smart Grid,” *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1934–1944, 2017.

[18] Y. Xue, Y.-A. Tan, C. Liang, C. Zhang, and J. Zheng, “An optimized data hiding scheme for Deflate codes,” *Soft Computing*, pp. 1–11, 2017.

[19] Z. Sun, Q. Zhang, Y. Li, and Y. Tan, “DPPDL: a Dynamic Partial-Parallel Data Layout for Green Video Surveillance Storage,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 1, pp. 193–205, 2018.

[20] H. Zhu, Y.-A. Tan, X. Zhang, L. Zhu, C. Zhang, and J. Zheng, “A round-optimal lattice-based blind signature scheme for cloud services,” *Future Generation Computer Systems*, vol. 73, pp. 106–114, 2017.

[21] X. Zhang, Y. Tan, C. Liang, Y. Li, and J. Li, “A Covert Channel Over VoLTE via Adjusting Silence Periods,” *IEEE Access*, vol. 6, pp. 9292–9302, 2018.

[22] Q. Lin, H. Yan, Z. Huang, W. Chen, J. Shen, and Y. Tang, “An ID-based linearly homomorphic signature scheme and its application in blockchain,” *IEEE Access*, vol. PP, no. 99, pp. 1-1, 2018.

[23] Y. Li, G. Wang, L. Nie, Q. Wang, and W. Tan, “Distance metric optimization driven convolutional neural network for age invariant face recognition,” *Pattern Recognition*, vol. 75, pp. 51–62, 2018.

[24] Z. Liu, Z. Wu, T. Li, J. Li, and C. Shen, “GMM and CNN Hybrid Method for Short Utterance Speaker Recognition,” *IEEE Transactions on Industrial Informatics*, vol. PP, no. 99, pp. 1-1, 2018.

[25] J. Xie, Y.-P. Hu, J.-T. Gao, and W. Gao, “Efficient identity-based signature over NTRU lattice,” *Frontiers of Information Technology and Electronic Engineering*, vol. 17, no. 2, pp. 135–142, 2016.

[26] S. Beura, M. Behera, and A. K. Tripathy, “Secured proxy blind signature scheme based on dlp with minimum computation cost,” *International Journal of Computer Science and Information Technologies*, vol. 2, no. 2, pp. 808–811, 2011.

[27] H. M. Sun, B. T. Hsieh, and S. M. Tseng, “On the security of some proxy blind signature schemes,” *The Journal of Systems and Software*, vol. 74, no. 3, pp. 297–302, 2005.

[28] C.-H. Wang and M.-Z. Liao, “Security Analysis and Enhanced Construction on ECDLP-Based Proxy Blind Signature Scheme,” *International Journal of e-Education, e-Business, e-Management and e-Learning*, vol. 4, no. 1, pp. 47–51, 2014.

[29] J. Zheng, Y.-a. Tan, Q. Zhang, X. Zhang, L. Zhu, and Q. Zhang, “Cross-cluster asymmetric group key agreement for wireless sensor networks,” *Science China Information Sciences*, vol. 61, no. 4, pp. 048103:1–048103:3, 2018.

[30] Z. Tan, “Efficient pairing-free provably secure identity-based proxy blind signature scheme,” *Security and Communication Networks*, vol. 6, no. 5, pp. 593–601, 2013.

[31] F. Zhang, R. Safavi-Naini, and C.-Y. Lin, “New proxy signature, proxy blind signature and proxy ring signature schemes from bilinear pairing,” in *Cryptology ePrint Archive, Report 2003/104*, 2003, https://eprint.iacr.org/2003/104.

[32] P. Li, J. Li, Z. Huang et al., “Multi-key privacy-preserving deep learning in cloud computing,” *Future Generation Computer Systems*, vol. 74, pp. 76–85, 2017.

[33] L. Zhang and Y. Ma, “A lattice-based identity-based proxy blind signature scheme in the standard model,” *Mathematical Problems in Engineering*, vol. 2014, Article ID 307637, 6 pages, 2014.

[34] N. Tahat and E. E. Abdallah, “A proxy partially blind signature approach using elliptic curve cryptosystem,” *International Journal of Mathematics in Operational Research*, vol. 8, no. 1, pp. 87–95, 2016.

[35] S. Mohapatra, S. Mohanty, A. Hota, and S. Pattanayak, “Data Security Enhancement in Cloud Computing using Proxy Blind Signature,” *International Journal of Computer Applications*, vol. 161, no. 2, pp. 27–31, 2017.

[36] K. Gu, W. Jia, Y. Deng, and X. Nie, “Secure and efficient multi-proxy signature scheme in the standard model,” *Journal of Electronics*, vol. 25, no. 1, pp. 93–99, 2016.

[37] X. Yang, C. Wang, L. Zhang, and J. Qiu, “On-line/off-line threshold proxy re-signatures,” *Chinese Journal of Electronics*, vol. 23, no. 2, pp. 248–253, 2014.

[38] X. Zhang, Y.-A. Tan, Y. Xue et al., “Cryptographic key protection against FROST for mobile devices,” *Cluster Computing*, vol. 20, no. 3, pp. 2393–2402, 2017.

[39] C. Gao, Q. Cheng, X. Li, and S. Xia, “Cloud-assisted privacy-preserving profile-matching scheme under multiple keys in mobile social network,” *Cluster Computing*, pp. 1–9, 2018.

[40] T. Peng, Q. Liu, D. Meng, and G. Wang, “Collaborative trajectory privacy preserving scheme in location-based services,” *Information Sciences*, vol. 387, pp. 165–179, 2017.

[41] Z. Chen, L. Peng, C. Gao, B. Yang, Y. Chen, and J. Li, “Flexible neural trees based early stage identification for IP traffic,” *Soft Computing*, vol. 21, no. 8, pp. 2035–2046, 2017.

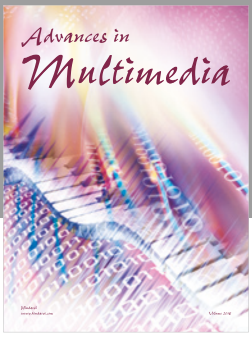
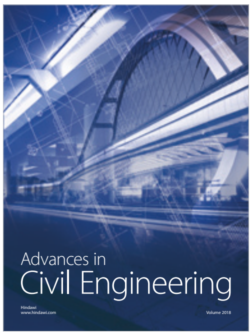
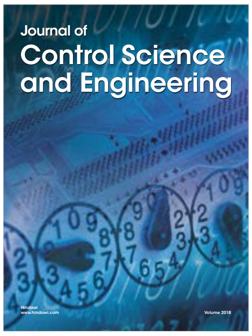
[42] J. Shen, Z. Gui, S. Ji, J. Shen, H. Tan, and Y. Tang, “Cloud-aided lightweight certificateless authentication protocol with anonymity for wireless body area networks,” *Journal of Network and Computer Applications*, vol. 106, pp. 117–123, 2018.

[43] X. J. Du, M. Guizani, Y. Xiao, and H.-H. Chen, “Transactions papers a routing-driven Elliptic Curve Cryptography based key management scheme for Heterogeneous Sensor Networks,” *IEEE Transactions on Wireless Communications*, vol. 8, no. 3, pp. 1223–1229, 2009.

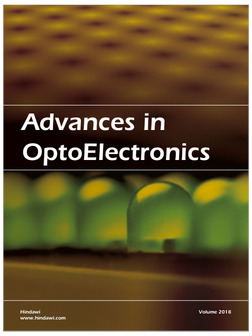
Journal of
Engineering

The Scientific
World Journal

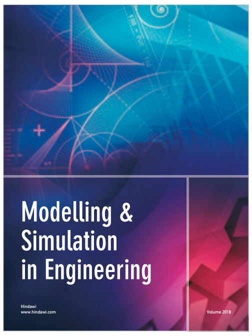
International Journal of
Rotating
Machinery

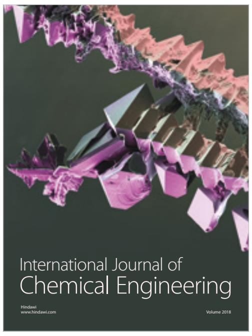
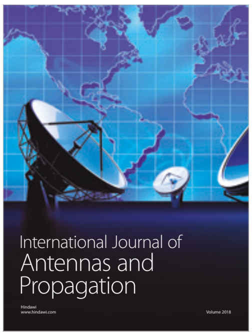
Journal of
Sensors

Advances in
Multimedia

Advances in
Civil Engineering

Journal of
Control Science
and Engineering

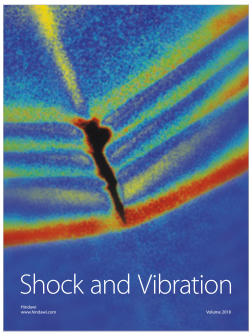
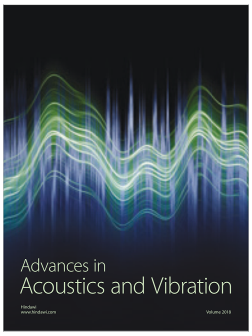
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

Advances in
OptoElectronics

VLSI Design

International Journal of
Navigation and
Observation

Modelling &
Simulation
in Engineering

International Journal of
Aerospace
Engineering

International Journal of
Chemical Engineering

International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components

Shock and Vibration

Advances in
Acoustics and Vibration