Singapore Management University

Institutional Knowledge at Singapore Management University

Research Collection School Of Computing and Information Systems | School of Computing and Information Systems

1-2018

Strong identity-based proxy signature schemes, revisited

Weiwei LIU

Yi MU

Guomin YANG

Yangguang TIAN
Singapore Management University, ygtian@smu.edu.sg

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research

 Part of the Information Security Commons

Citation
1

This Journal Article is brought to you for free and open access by the School of Computing and Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Computing and Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email cherylds@smu.edu.sg.

Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 6925019, 11 pages
https://doi.org/10.1155/2018/6925019

Research Article

Strong Identity-Based Proxy Signature Schemes, Revisited

Weiwei Liu[1], **Yi Mu**[2], **Guomin Yang**[2] **and Yangguang Tian**[3]

[1]*School of Mathematics and Statistics, North China University of Water Resources and Electric Power, Zhengzhou 450046, Henan, China*
[2]*Institute of Cybersecurity and Cryptology, School of Computing and Information Technology, University of Wollongong, NSW 2522, Australia*
[3]*School of Information Systems, Singapore Management University, Singapore*

Correspondence should be addressed to Weiwei Liu; liuweiwei@ncwu.edu.cn

Received 7 March 2018; Revised 30 May 2018; Accepted 14 June 2018; Published 6 August 2018

Academic Editor: Kim-Kwang Raymond Choo

Copyright © 2018 Weiwei Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Proxy signature is a useful cryptographic primitive that has been widely used in many applications. It has attracted a lot of attention since it was introduced. There have been lots of works in constructing efficient and secure proxy signature schemes. In this paper, we identify a new attack that has been neglected by many existing proven secure proxy signature schemes. We demonstrate this attack by launching it against an identity-based proxy signature scheme which is proven secure. We then propose one method that can effectively prevent this attack. The weakness in some other proxy signature schemes can also be fixed by applying the same method.

1. Introduction

Proxy signature is a special type of digital signature which allows one user (original signer) to delegate his/her signing right to another signer (proxy signer). The latter can then issue signatures on behalf of the former. The corresponding proxy signature can be verified by the public that it is indeed generated by the proxy signer with proper delegation from the original signer [1, 2]. Proxy signature has been found useful in many applications, such as distributed computing [3], electronic commerce [4], mobile agents [5], and grid computing [6]. It is worth noticing that proxy signature can also serve as a useful tool in Internet of things (IoT), since most of the RFID tags in IoT only have limited storage and computing ability. For those operations involving a large amount of computation, those tags can authorize the tag readers with strong computing ability to perform those operations with the help of a proxy signature scheme [7, 8].

The concept of proxy signature was introduced by Mambo, Usuda, and Okamoto in 1996 [9]. They presented three different types of proxy signature, namely, full delegation, partial delegation, and delegation by warrant in their seminal work. Shortly after Mambo et al.'s work, Kim et al. [10] proposed a new type of proxy signature combing partial delegation and warrant. They demonstrated that schemes combining partial delegation and warrant can provide a higher level of security than schemes based on partial delegation or warrant separately. Since then, proxy signature has been extensively researched in different settings, such as blind proxy signature [11], anonymous proxy signature [12], and identity-based proxy signature [13].

These delegation-by-warrant proxy signature schemes can be further classified into two categories according to whether the proxy signature is generated by the proxy signer using his own private key or not. In the first type, the proxy signer generates a new proxy signing key using the delegation information and his own private key. The proxy signatures are generated under the new proxy signing key. The proxy signature schemes in [5, 14–17] fall into the first type. In the second type, the proxy signer issues a proxy signature using his own private key. The proxy signatures are essentially combinations of the original signer's signature on the warrant and the proxy signer's signature on the message. Such proxy signature schemes could be found in [13, 18–21].

On the security modelling of proxy signature, Boldyreva et al. [22] proposed a comprehensive security model for

the delegation-by-warrant proxy signature, where an original signer can also perform self-delegation. Malkin et al. [23] extended the security model to allow fully hierarchical proxy signatures. They also proved that proxy signatures are essentially equivalent to key-insulated signatures. The security model proposed in [22, 23] is in the registered key model, which means the adversary has to submit every public and private key pair in the security game except the challenge one. Later, Schuldt et al. [24] proposed an enhanced security model for proxy signature by allowing the adversary to query arbitrary proxy signing keys. Roughly speaking, a secure proxy signature scheme should satisfy the following requirements.

(i) **Verifiability**: given a proxy signature, a verifier can be convinced that the proxy signature is indeed a valid signature generated by the proxy signer with proper delegation from an original signer on the signed message.

(ii) **Identifiability**: given a proxy signature, a verifier is able to determine the identities of the corresponding original signer and proxy signer.

(iii) **Unforgeability**: no one, except the designated proxy signer, can create a valid proxy signature.

(iv) **Untenability**: a proxy signer cannot deny at a later time on a proxy signature that he has created before.

(v) **Prevention of misuse**: it is required in the first type of proxy signature schemes that the proxy signing key cannot be used for purposes other than creating proxy signatures. Once misused, the identity of the misbehaving proxy signer can be determined explicitly.

1.1. Our Contribution. We revisit proxy signature and show an attack that has been neglected by the second type of proxy signature schemes [13, 18–21] that have been proven secure. In these schemes, a proxy signature is essentially the combination of the original signer's standard signature on a warrant and the proxy signer's standard signature on a message. In the security analysis, it is assumed that an adversary has access to the original signer and proxy signer's standard signature oracles. We show that, under such a circumstance, some proxy signature schemes [13, 18–21] that have been previously proved secure are in fact not secure.

We demonstrate a new attack by launching it against an identity-based proxy signature scheme [13] that has been proven secure. We show that a malicious adversary can create a proxy signature on a message, if he has access to the standard signature of the original signer and proxy signer, which is as defined in the security models in [13, 18]. Thus, these proxy signature schemes [13, 18–21], which we believe is not a complete list, are in fact not secure. We propose an efficient solution by revising the identity-based proxy signature scheme [13] to thwart this attack. It is worth noticing that the same method can also be applied to [18–21] to resist this attack.

We have noticed there have been several works [5, 22] aiming to transform normal proxy signature schemes into strong ones. The authors in [22] suggested to add two different prepositive tags "00" and "11" to distinguish the signatures generated by the original signer and proxy signer. However, this simple solution cannot prevent the attack proposed in this paper according to the original security model in [13]. The adversaries are able to query any message of their choices. To stop the proxy signer from misusing the proxy signing key, the authors in [5] classified existing proxy signature schemes into strong and weak ones and proposed one method to transform weak proxy signature schemes into strong ones. However, as have been mentioned above, their method is only applicable when a proxy signature is generated from a proxy signing key which is created by the proxy signer using the delegation information and his own private key. Therefore, the method proposed in [5] is not suitable for the scenarios discussed in this paper.

Paper Organization. The rest of the paper is organized as follows. We introduce some preliminaries in Section 2. Then we present a new attack in some proxy signature schemes in Section 3 by attacking an identity-based proxy signature scheme. The security model for proxy signature that captures the attack is presented in Section 4. We then revise the identity-based proxy signature scheme in Section 5. The security proof and efficiency analysis are presented in Section 6 and the paper is concluded in Section 7.

2. Preliminaries

In this section, we introduce some preliminaries used throughout this paper.

2.1. Bilinear Map. Let $\mathbb{G}_1$, $\mathbb{G}_2$ be two cyclic groups of prime order q and P a generator of $\mathbb{G}_1$. The $e : \mathbb{G}_1 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_2$ is said to be an admissible bilinear map if the following conditions hold:

(i) Bilinearity: $e(aP_1, bP_2) = e(abP_1, P_2) = e(P_1, abP_2)$ for all $P_1, P_2 \in \mathbb{G}_1$ and $a, b \in_R \mathbb{Z}_q$.

(ii) Nondegeneracy: there exists $P_1, P_2 \in \mathbb{G}_1$ such that $e(P_1, P_2) \neq 1_{\mathbb{G}_2}$.

(iii) Computability: there is an efficient algorithm to compute $e(P_1, P_2)$ for all $P_1, P_2 \in \mathbb{G}_1$.

2.2. Complexity Assumption

Definition 1 (computational Diffie-Hellman (CDH) problem). Given $P, aP, bP \in \mathbb{G}_1$ for some random $a, b \in \mathbb{Z}_q$, compute $abP \in \mathbb{G}_1$. Define the success probability of a polynomial algorithm $\mathscr{A}$ in solving the CDH problem as

$$Succ^{CDH}_{\mathscr{A},\mathbb{G}_1}(\kappa) = \Pr\left[\mathscr{A}(P, aP, bP) = abP : a, b \in_R \mathbb{Z}_q\right] \quad (1)$$

where $\kappa = \log(q)$ is the security parameter. The CDH assumption states that, for any polynomial algorithm adversary $\mathscr{A}$, $Succ^{CDH}_{\mathscr{A},\mathbb{G}_1}(\kappa)$ is negligible in κ.

3. A New Attack in Some Proxy Signature Schemes

In this section, we present an attack that has been neglected by many existing proxy signature schemes [13, 18–21]. To better explain how an attacker works, we demonstrate this attack via a concrete example. Before we start to introduce the attack, we first review an identity-based proxy signature scheme proposed in [13].

3.1. An Identity-Based Proxy Signature Scheme

(1) **Setup**: let $e : \mathbb{G}_1 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_2$ be a bilinear pairing map, where $\mathbb{G}_1$ and $\mathbb{G}_2$ are of prime order q. Let P be a generator of $\mathbb{G}_1$. Choose a random number $s \in \mathbb{Z}_q^*$ and set $P_{pub} = sP$. Select three collision-resistant hash functions H_0, H_1, H_2 such that $H_0, H_1, H_2 : \{0,1\}^* \longrightarrow \mathbb{G}_1$. The system parameters $params = \{e, \mathbb{G}_1, \mathbb{G}_2, q, P_{pub}, H_0, H_1, H_2\}$, the master secret key $Msk = s$.

(2) **KeyExtract**: on input a user's identity ID, output the secret key for this identity $sk_{ID} = sH_0(ID)$.

(3) **StandardSign**: on input a message m, the standard signature on m under identity ID is $\sigma = (\sigma_1, \sigma_2)$ such that $\sigma_1 = sk_{ID} + rH_1(M)$ and $\sigma_2 = rP$, where $r \in \mathbb{Z}_q$.

(4) **StandardVer**: on input a standard signature $\sigma = (\sigma_1, \sigma_2)$ of message m under identity ID, output "1" if $e(\sigma_1, P) = e(H_0(ID), P_{pub})e(H_1(m), \sigma_2)$; otherwise, output "0".

(5) **DelegationGen**: let w be a warrant that includes the delegation information such as the identities of the original signer and the designated proxy signer, the delegation period, the types of messages that a proxy signer can sign, and so on. Then the original signer with identity ID_A generates the delegation information $\sigma_w = (\sigma_{W_1}, \sigma_{W_2})$ such that $\sigma_{W_1} = sk_{ID_A} + r_A H_1(m_w)$ and $\sigma_{W_2} = r_A P$, where $r_A \in \mathbb{Z}_q$. The original signer sends the delegation signing key σ_w to the proxy signer.

(6) **ProSign**: upon receiving the delegation information $\sigma_w = (\sigma_{W_1}, \sigma_{W_2})$ and w from the original signer, the proxy signer with identity ID_B generates a proxy signature $\sigma = (\sigma_{M_1}, \sigma_{M_2}, \sigma_{M_3})$ on a message m such that $\sigma_{M_1} = \sigma_{W_1} + sk_{ID_B} + r_B H_2(m)$, $\sigma_{M_2} = \sigma_{W_2}$, $\sigma_{M_3} = r_B P$.

(7) **ProVer**: on input the identities ID_A, ID_B of the original signer and proxy signer, a warrant $w \in \{0,1\}^*$ and a message $m \in \{0,1\}^*$ and the proxy signature $\sigma = (\sigma_{M_1}, \sigma_{M_2}, \sigma_{M_3})$, output "1" if

$$e\left(\sigma_{M_1}, P\right) = e\left(H_0\left(ID_A\right), P_{pub}\right) e\left(H_1\left(w\right), \sigma_{M_2}\right) \cdot e\left(H_0\left(ID_B\right), P_{pub}\right) e\left(H_2\left(m\right), \sigma_{M_3}\right). \quad (2)$$

Otherwise, output "0".

3.2. An Attack against the ID-Based Proxy Signature Scheme.

Wu et al.'s identity-based proxy signature scheme [13] is proven secure. However, we show below that if the original signer and proxy signer also use their private keys to generate standard signatures, which is just as defined in their security models, then their scheme could be broken by a malicious outsider attacker. Assume the identities of the original signer and proxy signer are ID_A, ID_B, respectively, in the security model in [13], three types of adversaries are defined, namely,

(i) $\mathcal{A}_I$, which is an outsider adversary that has knowledge of (ID_A, ID_B),

(ii) $\mathcal{A}_{II}$, which is a malicious proxy signer that has knowledge of (ID_A, ID_B, sk_{ID_B}),

(iii) $\mathcal{A}_{III}$, which is a malicious original signer that has knowledge of (ID_A, sk_{ID_A}, ID_B).

The original signer and proxy signer could use the same key pairs to generate normal signatures using the standard signature scheme introduced in [13]. Suppose $\mathcal{A}_I$ aims to generate a proxy signature $\sigma = (\sigma_{M_1}, \sigma_{M_2}, \sigma_{M_3})$ on a message m with a warrant w; it is worth noticing that $\mathcal{A}_I$ might obtain such a genius warrant w when verifying a valid proxy signature. Then $\mathcal{A}_I$ acts as follows:

(i) $\mathcal{A}_I$ requires a standard signature $(\sigma_{A_1}, \sigma_{A_2})$ on warrant w of the original signer with identity ID_A, where w is a warrant containing the delegation information. The original signer chooses a random $r_A \in \mathbb{Z}_q$ and generates the standard signature $(\sigma_{A_1}, \sigma_{A_2})$ such that $\sigma_{A_1} = sk_{ID_A} + r_A H_1(w)$ and $\sigma_{A_2} = r_A P$.

(ii) Upon receiving the standard signature $(\sigma_{A_1}, \sigma_{A_2})$ on w from the original signer. $\mathcal{A}_I$ aborts if $e(\sigma_{A_1}, P) \neq e(H_0(ID_A), P_{pub})e(H_1(w), \sigma_{A_2})$.

(iii) $\mathcal{A}_I$ requires a standard signature $(\sigma_{B_1}, \sigma_{B_2})$ on message $w \parallel m$ of the proxy signer with identity ID_B, where m is a message. The proxy signer chooses a random $r_B \in \mathbb{Z}_q$ and generates the standard signature $(\sigma_{B_1}, \sigma_{B_2})$ such that $\sigma_{B_1} = sk_{ID_B} + r_B H_2(w, m)$ and $\sigma_{B_2} = r_B P$.

(iv) Upon receiving the standard signature $(\sigma_{B_1}, \sigma_{B_2})$ on m from the proxy signer. $\mathcal{A}_I$ aborts if $e(\sigma_{B_1}, P) \neq e(H_0(ID_B), P_{pub})e(H_2(w, m), \sigma_{B_2})$.

(v) If both $(\sigma_{A_1}, \sigma_{A_2})$ and $(\sigma_{B_1}, \sigma_{B_2})$ are valid. $\mathcal{A}_I$ outputs a proxy signature $\sigma = (\sigma_{M_1}, \sigma_{M_2}, \sigma_{M_3})$ on message m with warrant w such that $\sigma_{M_1} = \sigma_{A_1} + \sigma_{B_1} = sk_{ID_A} + r_A H_1(w) + sk_{ID_B} + r_B H_2(w, m)$, $\sigma_{M_2} = \sigma_{A_2} = r_A P$ and $\sigma_{M_3} = \sigma_{B_2} = r_B P$.

It can be verified that $\sigma = (\sigma_{M_1}, \sigma_{M_2}, \sigma_{M_3})$ is a valid proxy signature. Thus, the proposed identity-based proxy signature is insecure, since given a proxy signature $\sigma = (\sigma_{M_1}, \sigma_{M_2}, \sigma_{M_3})$, it might come from a malicious adversary. The proposed attack is a practical attack since a malicious adversary could launch such an attack without notice of both the original signer and the proxy signer. Besides the scheme mentioned in this paper, we have found that the proxy signature schemes in [18–21] are also subjected to this attack.

4. Security Model for Proxy Signature

4.1. Malicious Attackers. We revise the security model for identity-based proxy signature defined in [13] to capture the new attack in this section. In the security model for proxy signature, the capability of an adversary is modelled by its ability to query different oracles. Before we formally define each adversarial game, we first introduce four types of oracle queries that will appear in the models:

(i) **Key extract query**: $\mathcal{A}$ can query an identity $ID \in \mathcal{ID}$, where $\mathcal{ID}$ represents the identity space, to the key extract oracle $\mathcal{O}_{KE}(\cdot)$. The corresponding key sk_{ID} is then generated and returned to $\mathcal{A}$.

(ii) **Original signer's standard signing query**: $\mathcal{A}$ can query the original signer's signing oracle $\mathcal{O}_{OS'S}(\cdot)$ with any warrant $w \in \mathcal{W}$ under the original signer's identity $ID \in \mathcal{ID}$, where $\mathcal{W}$ represents the warrant space. The private key sk_{ID} on identity ID is generated using the key extraction algorithm. The corresponding original signer's signature σ_o on warrant w is generated and returned to $\mathcal{A}$.

(iii) **Proxy signing query**: $\mathcal{A}$ can query the proxy signing oracle $\mathcal{O}_{PS}(\cdot)$ with any message $m \in \mathcal{M}$ with warrant $w \in \mathcal{W}$ of his choice under the original signer's identity ID_A and the proxy signer's identity ID_B such that $ID_A, ID_B \in \mathcal{ID}$, where $\mathcal{M}$ represents the message space. The private keys sk_{ID_A} and sk_{ID_B} on identities ID_A, ID_B are generated using the key extraction algorithm. A valid proxy signature on m is then generated and returned to $\mathcal{A}$.

(iv) **Proxy signer's signing query**: $\mathcal{A}$ can query the standard signature with any message $m \in \mathcal{M}$ of his choice to the proxy signer's standard signing oracle $\mathcal{O}_{PS'S}(\cdot)$. A valid standard signature of the proxy signer σ_p on m under the proxy signer's identity is then generated and returned to $\mathcal{A}$.

According to the information held by an attacker, three different types of adversaries are defined:

(1) $\mathcal{A}_I$: an outsider attacker who only has the identities of the original signer and the proxy signer that aims to forge a valid proxy signature.

(2) $\mathcal{A}_{II}$: a malicious proxy signer who possesses the private key sk_{ID_B} of the proxy signer and the identity of the original signer, and tries to forge a valid proxy signature σ without knowledge of the private key sk_{ID_A} of the original signer.

(3) $\mathcal{A}_{III}$: a malicious original signer that possesses the private key sk_{ID_A} of the original signer and the identity ID_B of the proxy signer, and tries to forge a valid proxy signature σ without knowing the private key sk_{ID_B} of the proxy signer.

4.2. Adversarial Game with a Malicious Outsider Adversary $\mathcal{A}_I$. We first define the adversarial game between a malicious outsider adversary $\mathcal{A}_I$ and a simulator $\mathcal{S}$ as follows:

(i) **Setup**: the simulator $\mathcal{S}$ runs **Setup** algorithm to generate the *params* and *MSK* and sends *params* to $\mathcal{A}_I$ as well as keeping *MSK* secret.

(ii) **Original signer's standard signing queries**: $\mathcal{A}_I$ can choose any warrant $w \in \mathcal{W}$ with the original signer's identity ID_A and queries the original signer's standard signing oracle $\mathcal{O}_{OS'S}$. $\mathcal{S}$ generates the private key sk_{ID_A} using the key extract algorithm $sk_{ID_A} \longleftarrow$ **KeyExtract**$(MSK, ID_A, params)$; then $\mathcal{S}$ generates the delegation information $\sigma_o \longleftarrow$ **StandardSign**$(sk_{ID_A}, w, params)$ and sends σ_o to $\mathcal{A}_I$.

(iii) **Proxy Signer's Standard Signature Queries**: $\mathcal{A}_I$ queries the proxy signer's standard signing oracle $\mathcal{O}_{PS'S}$ with a message $m \in \mathcal{M}$ of his choice under the proxy signer's identity $ID_B \in \mathcal{ID}$. $\mathcal{S}$ generates the private key sk_{ID_B} using the key extract algorithm $sk_{ID_B} \longleftarrow$ **KeyExtract**$(MSK, ID_B, params)$; then $\mathcal{S}$ generates the standard signature $s\sigma \longleftarrow$ **StandardSign**$(sk_{ID_B}, m, params)$ and sends $s\sigma$ to $\mathcal{A}_I$.

(iv) **Forgery Phase**: finally, $\mathcal{A}_I$ outputs a proxy signature σ^* on message M^* for a warrant W^* with the original signer's identity ID_A and the proxy signer's identity ID_B.

We say $\mathcal{A}_{II}$ wins the game if

(i) **ProVer**$(\sigma^*, ID_A, ID_B, W^*, M^*) = 1$;

(ii) (W^*, ID_A) has been queried to the original signer's standard signing oracle $\mathcal{O}_{OS'S}$;

(iii) (W^*, M^*, ID_B) has been queried to the proxy signer's standard signing oracle $\mathcal{O}_{PS'S}$.

Define the advantage of a malicious adversary $\mathcal{A}_I$ in winning the game as

$$Adv_{\mathcal{A}_I}(\kappa) = \Pr\left[\mathcal{A}_I \text{ Wins the game}\right]. \tag{3}$$

Definition 2. We say an identity-based proxy signature scheme is secure against an outsider adversary $\mathcal{A}_I$ if for any probabilistic polynomial time $\mathcal{A}_I$, $Adv_{\mathcal{A}_I}(\kappa)$ is negligible in κ.

4.3. Adversarial Game with a Malicious Proxy Signer $\mathcal{A}_{II}$. We first define the adversarial game between a malicious proxy signer $\mathcal{A}_{II}$ and a simulator $\mathcal{S}$ as follows:

(i) **Setup**: the simulator $\mathcal{S}$ runs **Setup** algorithm to generate the *params* and *MSK* and sends *params* to $\mathcal{A}_{II}$ as well as keeping *MSK* secret.

(ii) **Key extract queries**: $\mathcal{A}_{II}$ selects an identity ID such that $ID \in \mathcal{ID}$, the simulator $\mathcal{S}$ runs $sk_{ID} \longleftarrow$ **KeyExtract**$(MSK, ID, params)$ and returns sk_{ID} to $\mathcal{A}_{II}$.

(iii) **Original signer's standard signing queries**: $\mathcal{A}_{II}$ can choose any warrant $w \in \mathcal{W}$ with an identity $ID \in \mathcal{ID}$ and queries original signer's standard signing oracle $\mathcal{O}_{OS'S}$. $\mathcal{S}$ generates the private key sk_{ID} using the key extract algorithm $sk_{ID} \longleftarrow$ **KeyExtract**$(MSK, ID, params)$; then $\mathcal{S}$ generates the

original signer's standard signature $\sigma_o \longleftarrow$ **StandardSign**$(sk_{ID}, w, params)$ and sends σ_o to $\mathscr{A}_{II}$.

(iv) **Proxy signing queries**: $\mathscr{A}_{II}$ chooses a warrant $w \in \mathscr{W}$ and a message $m \in \mathscr{M}$ and queries the proxy signing oracle $\mathscr{O}_{PS}$ with the original signer's identity ID_1 and the proxy signer's identity ID_2. $\mathscr{S}$ generates

$$\begin{aligned} &sk_{ID_1}, sk_{ID_2} \\ &\quad \longleftarrow \textbf{KeyExtract}\left(MSK, ID_1, ID_2, params\right) \\ &\sigma_w \longleftarrow \textbf{DelegationGen}\left(sk_{ID_1}, w, params\right), \\ &\sigma \longleftarrow \textbf{ProSign}\left(\sigma_w, sk_{ID_2}, m, params\right) \end{aligned} \tag{4}$$

and returns σ to $\mathscr{A}_{II}$.

(v) **Forgery Phase**: finally, $\mathscr{A}$ outputs a proxy signature σ^* on message M^* for a warrant W^* with the original signer's identity ID_A and the proxy signer's identity ID_B.

We say $\mathscr{A}_{II}$ wins the game if

(i) **ProVer**$(\sigma^*, ID_A, ID_B, W^*, M^*) = 1$;

(ii) ID_A has not been queried to the key extraction oracle $\mathscr{O}_{KE}(\cdot)$;

(iii) (W^*, ID_A) has not been queried to the delegation oracle $\mathscr{O}_{DG}$;

(iv) (W^*, M^*, ID_A, ID_B) has not been queried to the proxy signing oracle $\mathscr{O}_{PS}$.

Define the advantage of a malicious adversary $\mathscr{A}_{II}$ in winning the game as

$$Adv_{\mathscr{A}_{II}}(\kappa) = \Pr\left[\mathscr{A}_{II} \text{ Wins the game}\right]. \tag{5}$$

Definition 3. We say an identity-based proxy signature scheme is secure against the $\mathscr{A}_{II}$ under chosen identity and warrant attacks if for any probabilistic polynomial time $\mathscr{A}_{II}$, $Adv_{\mathscr{A}_{II}}(\kappa)$ is negligible in κ.

4.4. Adversarial Game with Malicious Original Signer. The adversarial game between a malicious original signer $\mathscr{A}_{III}$ and a simulator $\mathscr{S}$ is defined as follows:

(i) **Setup**, **Key Extract Queries** and **Proxy Signing Queries** are the same as those in the adversarial game against a malicious proxy signer.

(ii) **Proxy Signer's Standard Signature Queries**: $\mathscr{A}_{III}$ queries the proxy signer's standard signing oracle $\mathscr{O}_{ps's}$ with a message $m \in \mathscr{M}$ of his choice under an identity $ID \in \mathscr{ID}$. $\mathscr{S}$ generates the private key sk_{ID} using the key extract algorithm $sk_{ID} \longleftarrow$ **KeyExtract**$(MSK, ID, params)$; then $\mathscr{S}$ generates the standard signature $\sigma_p \longleftarrow$ **StandardSign**$(sk_{ID}, m, params)$ and sends σ_p to $\mathscr{A}_{III}$.

(iii) **Forgery Phase**: finally, $\mathscr{A}_{III}$ outputs a proxy signature σ^* on message M^* for a warrant W^* with the original signer's identity ID_A and the proxy signer's identity ID_B.

We say $\mathscr{A}_{III}$ wins the game if

(i) **ProVer**$(\sigma^*, ID_A, ID_B, W^*, M^*) = 1$;

(ii) ID_B has not been queried to the key extraction oracle $\mathscr{O}_{KE}$;

(iii) (W^*, M^*, ID_B) has not been queried to the proxy signer's standard signing oracle $\mathscr{O}_{PS'S}$;

(iv) (W^*, M^*, ID_A, ID_B) has not been queried to the proxy signing oracle $\mathscr{O}_{PS}$.

Define the advantage of a malicious adversary $\mathscr{A}_{III}$ in winning the game as

$$Adv_{\mathscr{A}_{III}}(\kappa) = \Pr\left[\mathscr{A}_{III} \text{ Wins the game}\right]. \tag{6}$$

Definition 4. We say an identity-based proxy signature scheme is secure against the $\mathscr{A}_{III}$ under chosen identity and message attacks if for any probabilistic polynomial time $\mathscr{A}_{III}$, $Adv_{\mathscr{A}_{III}}(\kappa)$ is negligible in κ.

5. The Revised Identity-Based Proxy Signature Scheme

We present the revised ID-based proxy signature scheme that efficiently thwarts the proposed attack in this section.

(1) **Setup**: let $e : \mathbb{G}_1 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_2$ be a bilinear pairing map, where $\mathbb{G}_1$ and $\mathbb{G}_2$ are of prime order q. Let P be a generator of $\mathbb{G}_1$. Choose a random number $s \in \mathbb{Z}_q^*$ and set $P_{pub} = sP$. Select three collision-resistant hash functions H_0, H_1, H_2 such that $H_0, H_1, H_2 : \{0,1\}^* \longrightarrow \mathbb{G}_1$. The system parameters $params = \{e, \mathbb{G}_1, \mathbb{G}_2, q, P_{pub}, H_0, H_1, H_2\}$, the master secret key $Msk = s$.

(2) **KeyExtract**: on input a user's identity ID, output the secret key for this identity $sk_{ID} = sH_0(ID)$.

(3) **StandardSign**: on input a message m, the standard signature on m under identity ID is $\sigma = (\sigma_1, \sigma_2)$ such that $\sigma_1 = sk_{ID} + rH_1(m)$ and $\sigma_2 = rP$, where $r \in \mathbb{Z}_q^*$.

(4) **StandardVer**: on input a standard signature $\sigma = (\sigma_1, \sigma_2)$ of message m under identity ID, output "1" if $e(\sigma_1, P) = e(H_0(ID), P_{pub})e(H_1(m), \sigma_2)$; otherwise, output "0".

(5) **DelegationGen**: let w be a warrant that includes the delegation information such as the identities of the original signer and the designated proxy signer, the delegation period, the types of messages that a proxy signer can sign, and so on. Then the original signer with identity ID_A generates the delegation information $\sigma_w = (\sigma_{W_1}, \sigma_{W_2})$ such that $\sigma_{W_1} = sk_{ID_A} + r_A H_1(w)$ and $\sigma_{W_2} = r_A P$, where $r_A \in \mathbb{Z}_q$. The original signer sends the delegation information σ_w to the proxy signer.

(6) **ProSign**: upon receiving the delegation information $\sigma_w = (\sigma_{W_1}, \sigma_{W_2})$ and w from the original signer, the proxy signer with identity ID_B generates a proxy signature $\sigma = (\sigma_{M_1}, \sigma_{M_2}, \sigma_{M_3}$ on a message m such that $\sigma_{M_1} = \sigma_{W_1} + sk_{ID_B} + r_B H_2(w, m) + r_B H_1(w)$, $\sigma_{M_2} = \sigma_{W_2} + r_B P$, $\sigma_{M_3} = r_B P$.

(7) **ProVer**: on input the identities ID_A, ID_B of the original signer and proxy signer, a warrant w and a message m and the proxy signature $\sigma = (\sigma_{M_1}, \sigma_{M_2}, \sigma_{M_3})$, outputs "1" if $e(\sigma_{M_1}, P) = e(H_0(ID_A), P_{pub}) e(H_1(w)), \sigma_{M_2}) \cdot e(H_0(ID_B), P_{pub}) e(H_2(w, m), \sigma_{M_3})$. Otherwise, output "0".

6. Security Analysis

In this section, we analyse the security of the revised ID-based proxy signature scheme against $\mathscr{A}_I$, $\mathscr{A}_{II}$, and $\mathscr{A}_{III}$ adversaries.

Theorem 5. *The revised ID-based proxy signature scheme is secure against an outsider adversary $\mathscr{A}_I$ if the CDH assumption holds.*

Proof. The proof is by contradiction under the random oracle model. Suppose there exists an outsider adversary $\mathscr{A}_I$ that has a nonnegligible advantage ϵ in attacking the proposed scheme; then we can build another algorithm $\mathscr{B}$ that uses $\mathscr{A}_I$ to solve the CDH problem. Let $\mathbb{G}_1$ be a bilinear pairing group of prime order q; $\mathscr{B}$ is given $P, aP, bP \in \mathbb{G}_1$ which is a random instance of the CDH problem. Its goal is to compute abP. Algorithm $\mathscr{B}$ will simulate the challenger and interact with the forger $\mathscr{A}_I$ as described below.

(1) **Setup**: $\mathscr{B}$ selects a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_2$ where $\mathbb{G}_1$ and $\mathbb{G}_2$ are of prime order q. $\mathscr{B}$ chooses a generator P of $\mathbb{G}_1$. Let (P, aP, bP) be the inputs of the CDH problem. $\mathscr{B}$ sets the master public key $P_{pub} = sP$, where $s \in \mathbb{Z}_q^*$. $\mathscr{B}$ selects three collision-resistant hash functions $H_0, H_1, H_2 : \{0,1\}^* \longrightarrow \mathbb{G}_1$. $\mathscr{B}$ sends $(e, \mathbb{G}_1, \mathbb{G}_2, q, P, P_{pub}, H_0, H_1, H_2)$ to $\mathscr{A}_{II}$.

(2) **Hash queries**: in the security proof, the hash functions H_0, H_1, H_2 are modelled as random oracles. We regard the identity, warrant, and message queries as H_0, H_1, and H_2 queries, respectively. Assume $\mathscr{B}$ keeps hash tables T_0, T_1, and T_2 for these queries.

(a) H_0 **Query**: for each query on identity ID_i, if ID_i has existed in T_0, the same value $H_0(ID_i)$ is returned to $\mathscr{A}_{II}$. Otherwise, $\mathscr{B}$ chooses a random $c_i \in \mathbb{Z}_q$ and sets $H_0(ID_i) = c_i P$. $\mathscr{B}$ sends $c_i P$ to $\mathscr{A}_I$ as well as stores $(ID_i, c_i, H_0(ID_i))$ to T_0.

(b) H_1 **Query**: assume $\mathscr{A}_I$ makes q_{H_1} warrant queries; $\mathscr{B}$ selects a random number $\beta \in (1, q_{H_1})$, for each query on warrant w_i such that $1 \le i \ne \beta \le q_{H_1}$; if w_i has existed in T_1, the same value $H_1(w_i)$ is returned to $\mathscr{A}_I$. Otherwise,

(i) if $w_i \ne w_\beta$, $\mathscr{B}$ chooses a random $k_i \in \mathbb{Z}_q$ and sets $H_1(w_\beta) = k_i P$. $\mathscr{B}$ sends $H_1(w_\beta)$ to $\mathscr{A}_I$ as well as storing $(w_\beta, k_i, H_1(w_\beta))$ to T_1.

(ii) If $w_i = w_\beta$, $\mathscr{B}$ sets $H_1(w_\beta) = aP$. $\mathscr{B}$ sends $H_1(w_\beta)$ to $\mathscr{A}_I$.

(c) H_2 **Query**: for each query on message m_i accompanying with a warrant w_i, if $H_2(w_i, m_i)$ has existed in T_2, the same value $H_2(w_i, m_i)$ is returned to $\mathscr{A}_I$. Otherwise, $\mathscr{B}$ chooses a random $u_i \in \mathbb{Z}_q$ and sets $H_2(w_i, m_i) = u_i P$. $\mathscr{B}$ sends $H_2(w_i, m_i)$ to $\mathscr{A}_I$ as well as storing $((w_i, m_i), u_i, H_2(w_i, \mathrm{m}_i))$ to T_2.

(3) **Original signer's standard signing queries**: $\mathscr{A}_I$ can query the original signer's standard signature on a warrant w_i. Assume $\mathscr{A}_I$ makes $q_{os's}$ queries with the original signer's identity ID_A, for each query on w_i, assume $H_0(ID_A)$ and $H_1(w_i)$ have existed in T_0 and T_1; if they are not the cases, $\mathscr{B}$ performs the above algorithms to assign values for $H_0(ID_A)$ and $H_1(w_i)$. Assume $H_0(ID_A) = c_A P$, $\mathscr{B}$ simulates as follows:

(i) If $w_i \ne w_\beta$, assume $H_1(w_i) = k_i P$; then $\mathscr{B}$ chooses randomly $r_{A_i} \in \mathbb{Z}_q$ and sets $\sigma_{w_i} = (\sigma_{w_{i1}}, \sigma_{w_{i2}})$ such that $\sigma_{w_{i1}} = c_A sP + r_{A_i} k_i P = sH_0(ID_A) + r_{A_i} H_1(w_i)$ and $\sigma_{w_{i2}} = r_{A_i} P$.

(ii) If $w_i = w_\beta$, then $\mathscr{B}$ chooses randomly $r_{A_\beta} \in \mathbb{Z}_q$ and sets $\sigma_\beta = (\sigma_{w_{\beta 1}}, \sigma_{2_{\beta 2}})$ such that $\sigma_{w_{\beta 1}} = c_A sP + r_{A_\beta} bP = sH_0(ID_A) + r_{A_\beta} H_1(w_\beta)$ and $\sigma_{w_{\beta 2}} = r_{A_\beta} P$.

(4) **Proxy signer's standard signing queries**: assume $\mathscr{A}_I$ makes $q_{ps's}$ standard signature queries under the proxy signer's identity ID_B. For each query on $M_i = w_i \parallel m_i$, assume $H_0(ID_B)$ and $H_2(M_i)$ have existed in T_0 and T_2; if they are not the cases, $\mathscr{B}$ performs the above algorithms to assign values for $H_0(ID_A)$ and $H_2(M_i)$. Assume $H_0(ID_B) = c_B P$; $\mathscr{B}$ chooses a number $\delta \in (1, q_{ps's})$ and simulates as follows:

(i) If $M_i \ne M_\delta$, assume $H_2(M_2) = u_i P$; then $\mathscr{B}$ chooses randomly $r_{B_i} \in \mathbb{Z}_q$ and sets $\sigma_{p_i} = (\sigma_{p_{i1}}, \sigma_{p_{i2}})$ such that $\sigma_{p_{i1}} = c_B sP + r_{B_i} k_i P = sH_0(ID_B) + r_{B_i} H_2(M_i)$ and $\sigma_{p_{i1}} = r_{B_i} P$.

(ii) If $M_i = M_\delta$, assume $H_2(M_\delta) = u_\delta P$; then $\mathscr{B}$ sets $dsk_\delta = (\sigma_{B1_\delta}, \sigma_{B2_\delta})$ such that $\sigma_{B1_\delta} = c_B sP + bu_\delta P = sH_0(ID_B) + bH_2(M_\delta)$ and $\sigma_{B2_i} = bP$.

(5) **Forgery**: assume $\mathscr{A}_I$ outputs a valid proxy signature $\sigma^* = (\sigma^*_{M_1}, \sigma^*_{M_2}, \sigma^*_{M_3})$ on message M^* under a warrant W^* with the proxy signer's identity ID_A and the proxy signer's identity ID_B. Besides,

(i) (ID_A, W^*) has been queried in the original signer's standard signing queries;

(ii) (ID_B, W^*, M^*) has been queried in the proxy signer's standard signing queries.

If $W^* \ne w_\beta$ or $M^* \ne M_\delta$, $\mathscr{B}$ will abort. Otherwise, given the forged proxy signature $\sigma^* = (\sigma^*_{M_1}, \sigma^*_{M_2}, \sigma^*_{M_3})$. $\mathscr{B}$ can solve the CDH problem

$$abP = \sigma_{M_1^*} - \sigma_{A1_\beta} - \sigma_{B1_\delta} \tag{7}$$

$\mathscr{B}$ will not abort when $W^* = w_\beta$ and $M^* = M_\delta$. Thus, if there exists an outsider adversary $\mathscr{A}_I$ that has a nonnegligible probability ϵ in breading the proposed identity-based proxy signature scheme, then there exists another probabilistic polynomial time algorithm $\mathscr{B}$ that has a probability

$$Succ^{CDH}_{\mathscr{B},\mathbb{G}_1} = \frac{\epsilon}{q_{os's} \cdot q_{ps's}} \tag{8}$$

which is nonnegligible. Thus, we reach a contradiction. □

Theorem 6. *The revised ID-based proxy signature scheme is secure against the $\mathscr{A}_{II}$ chosen identity and chosen warrant attacks if the CDH assumption holds.*

Proof. Let us recall the definition of $\mathscr{A}_{II}$; $\mathscr{A}_{II}$ is a malicious proxy signer possessing the private key of the proxy signer. With this in mind, the simulation is as follows:

(1) **Setup**: $\mathscr{B}$ selects a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_2$, where $\mathbb{G}_1$ and $\mathbb{G}_2$ are of prime order q. $\mathscr{B}$ chooses a generator P of $\mathbb{G}_1$. Let (P, aP, bP) be the inputs of the CDH problem. $\mathscr{B}$ sets the master public key $P_{pub} = aP$. $\mathscr{B}$ selects three collision-resistant hash functions $H_0, H_1, H_2 : \{0,1\}^* \longrightarrow \mathbb{G}_1$. $\mathscr{B}$ sends $(e, \mathbb{G}_1, \mathbb{G}_2, q, P, P_{pub}, H_0, H_1, H_2)$ to $\mathscr{A}_{II}$.

(2) **Hash queries**: regard the identity, warrant, and message queries as H_0, H_1, and H_2 queries, respectively. $\mathscr{B}$ keeps hash tables T_0, T_1, and T_2 for these queries.

(a) H_0 **Query**: assume $\mathscr{A}_{II}$ makes q_{H_0} identity queries, choose $\alpha \in (1, q_{H_0})$, for each query on identity ID_i such that $1 \le i \ne \alpha \le q_{H_0}$, if ID_i has existed in T_0, the same value $H_0(ID_i)$ is returned to $\mathscr{A}_{II}$. Otherwise,

(i) If $i \ne \alpha$, $\mathscr{B}$ chooses a random $c_i \in \mathbb{Z}_q$ and sets $H_0(ID_i) = c_iP$. $\mathscr{B}$ sends c_iP to $\mathscr{A}_{II}$ as well as storing $(ID_i, c_i, H_0(ID_i))$ to T_0.

(ii) If $i = \alpha$, $\mathscr{B}$ sets $H_0(ID_\alpha) = bP + c_\alpha P$, where $c_\alpha \in \mathbb{Z}_q$ and returns $H_0(ID_I)$ to $\mathscr{A}_{II}$. $\mathscr{B}$ adds $(ID_\alpha, c_\alpha, H_0(ID_\alpha))$ to T_0.

(b) H_1 **Query**: assume $\mathscr{A}_{II}$ makes q_{H_1} warrant queries; $\mathscr{B}$ selects a random number $\beta \in (1, q_{H_1})$, for each query on warrant w_i such that $1 \le i \ne \beta \le q_{H_1}$, if w_i has existed in T_1, the same value $H_1(w_i)$ is returned to $\mathscr{A}_{II}$. Otherwise,

(i) if $w_i \ne w_\beta|_{ID_\alpha \longrightarrow o}$, which means ID_α is included in w_i and the user with identity ID_α plays the role of original signer in the system. $\mathscr{B}$ chooses a random $k_i \in \mathbb{Z}_q$ and sets $H_1(w_i) = k_iP - bP$. $\mathscr{B}$ sends $H_1(w_i)$ to $\mathscr{A}_{II}$ as well as storing $(w_i, b_i, H_1(w_i))$ to T_1;

(ii) if $w_i \ne w_\beta|_{ID_\alpha \longrightarrow p}$, which means ID_α is included in w_i and the user with identity ID_α plays the role of proxy signer in the system. $\mathscr{B}$ chooses a random $k_i \in \mathbb{Z}_q$ and sets $H_1(w_i) = k_iP$. $\mathscr{B}$ sends $H_1(w_i)$ to $\mathscr{A}_{II}$ as well as stores $(w_i, k_i, H_1(w_i))$ to T_1;

(iii) if $w_i = w_\beta$, $\mathscr{B}$ chooses a random $k_i \in \mathbb{Z}_q$ and sets $H_1(w_\beta) = k_iP$. $\mathscr{B}$ sends $H_1(w_\beta)$ to $\mathscr{A}_{II}$ as well as storing $(w_\beta, k_i, H_1(w_\beta))$ to T_1.

(c) H_2 **Query**: assume $\mathscr{A}_{II}$ makes q_{H_2} message queries, $\mathscr{B}$ selects a random number $\delta \in (1, q_{H_1})$, for each query on message m_i accompanying with a warrant w_i such that $1 \le i \ne \delta \le q_{H_2}$, if $H_2(w_i, m_i)$ has existed in T_2, the same value $H_2(w_i, m_i)$ is returned to $\mathscr{A}_{II}$. Otherwise,

(i) if $w_i \ne w_\beta, m_i \ne m_\delta$, $\mathscr{B}$ chooses a random $u_i \in \mathbb{Z}_q$ and sets $H_2(w_i, m_i) = u_iP + aP$. $\mathscr{B}$ sends $H_2(w_i, m_i)$ to $\mathscr{A}_{II}$ as well as storing $((w_i, m_i), c_i, H_2(w_i, m_i))$ to T_2;

(ii) if $w_i = w_\beta, m_i \ne m_\delta$, the same as the case when $w_i \ne w_\beta, m_i \ne m_\delta$;

(iii) if $w_i \ne w_\beta, m_i = m_\delta$, the same as the case when $w_i \ne w_\beta, m_i \ne m_\delta$;

(iv) if $w_i = w_\beta, m_i = m_\delta$, $\mathscr{B}$ chooses a random $u_i \in \mathbb{Z}_q$ and sets $H_2(w_\beta, m_\delta) = u_iP$. $\mathscr{B}$ sends $H_2(w_\beta, m_\beta)$ to $\mathscr{A}_{II}$ as well as storing $((w_\beta, m_\delta), u_i, H_2(w_\beta, m_\delta))$ to T_2.

(3) **Key extraction queries**: $\mathscr{A}_{II}$ can make key extraction queries on any identity $ID \in \mathcal{ID}$ such that $ID \ne ID_\alpha$. If $\mathscr{A}_{II}$ makes key extraction query on identity ID_α, $\mathscr{B}$ just terminates the simulation and reports a failure. Assume $\mathscr{A}_{II}$ makes q_k key extractions queries, for each query on identity ID_i for $1 \le i \le q_k$.

(i) If ID_i has existed in table T_0, assume $H_0(ID_i) = c_iP$; then $\mathscr{B}$ returns $sk_{ID_i} = c_iaP = aH_0(ID_i)$ to $\mathscr{A}_{II}$.

(ii) Otherwise, $\mathscr{B}$ chooses a random $c_i \in \mathbb{Z}_q$ and sets $H_0(ID_i) = c_iP$. $\mathscr{B}$ returns $sk_{ID_i} = c_iaP$ to $\mathscr{A}_{II}$ and adds $(ID_i, c_i, H_0(ID_i))$ to T_0.

(4) **Original signer's standard signing queries**: $\mathscr{A}_{II}$ can query original signer's standard signature on a warrant $w_i \in \mathcal{W}$ under an identity $ID_i \in \mathcal{ID}$. Assume $\mathscr{A}_{II}$ makes $q_{os's}$ original signer's standard signing queries. For each query, assume ID_i and w_i have been submitted to the H_0 and H_1 queries, respectively. If they are not the cases, $\mathscr{B}$ performs the above algorithms to set values for $H_0(ID_i)$ and $H_1(w_i)$; then $\mathscr{B}$ simulates σ_{w_i} as follows:

(i) If $ID_i \ne ID_\alpha$ and $w_i \ne w_\beta|_{ID_\alpha \longrightarrow o}$, assume $H_0(ID_i) = c_iP$ and $H_1(w_i) = k_iP - bP$, respectively; then $\mathscr{B}$ chooses a random $r_i \in \mathbb{Z}_q$ and returns the original signer's standard signature $\sigma_{w_i} = (\sigma w_{i1}, \sigma w_{i2})$ such that $\sigma w_{i1} = c_iP_{pub} + r_i(k_iP - bP) = sk_{ID_i} + r_iH_1(w_i)$ and $\sigma w_{i2} = r_iP$ and to $\mathscr{A}_{II}$.

(ii) If $ID_i \ne ID_\alpha$ and $w_i \ne w_\beta|_{ID_\alpha \longrightarrow p}$, assume $H_0(ID_i) = c_iP$ and $H_1(w_i) = k_iP$, respectively; then $\mathscr{B}$ chooses a random $r_i \in \mathbb{Z}_q$ and returns original signer's standard signature $\sigma_{w_i} = (\sigma w_{i1}, \sigma w_{i2})$ such that $\sigma w_{i1} = c_iP_{pub} + r_ik_iP = sk_{ID_i} + r_iH_1(w_i)$ and $\sigma w_{i2} = r_iP$ to $\mathscr{A}_{II}$.

(iii) If $ID_i = ID_\alpha$ and $w_i \neq w_\beta|_{ID_\alpha \longrightarrow o}$, assume $H_0(ID_i) = bP + c_iP$ and $H_1(w_i) = k_iP - bP$, respectively; then $\mathscr{B}$ simulates the original signer's standard signature $\sigma_{w_i} = (\sigma w_{i1}, \sigma w_{i2})$ by setting $\sigma w_{i2} = r_iP = l_iP + aP$, where $l_i \in_R \mathbb{Z}_q^*$ and $\sigma w_{i1} = (c_i + k_i)P_{pub} + k_il_iP - l_ibP$. It can be verified that $(\sigma w_{i1}, \sigma w_{i2})$ is a correct simulation since

$$\begin{aligned}\sigma w_{i1} &= (c_i + k_i) P_{pub} + k_il_iP - l_ibP \\ &= abP + c_iaP + k_iaP + k_il_iP - l_ibP - abP \\ &= a(c_iP + bP) + (a + l_i)(k_iP - bP) \\ &= aH_0(ID_\alpha) + r_iH_1(w_i)\end{aligned} \tag{9}$$

(iv) If $ID_i = ID_\alpha$ and $w_i \neq w_\beta|_{ID_\alpha \longrightarrow p}$, since we do not consider self-delegation in our scheme, then $\mathscr{B}$ just terminates the simulation and reports failure.

(v) If $ID_i = ID_\alpha$ and $w_i = w_\beta$, $\mathscr{B}$ terminates the simulation and reports failure.

(5) **Proxy signing queries**: $\mathscr{A}_{II}$ can query a proxy signature on a message $m_i \in \mathscr{M}$ under a warrant $w_i \in \mathscr{W}$ with the proxy signer's identity ID_{1_i} and the original signer's identity ID_{2_i} such that $ID_{1_i}, ID_{2_i} \in \mathscr{ID}$. Assume ID_{1_i}, ID_{2_i} have been submitted to the H_0 query and w_i and $w_i \parallel m_i$ have been submitted to the H_1 and H_2 queries, respectively. If they are not the cases, the above algorithms will be performed to assign new values $H_0(ID_{1_i})$, $H_0(ID_{2_i})$, $H_1(w_i)$, and $H_2(w_i, m_i)$. Assume $\mathscr{A}_{II}$ makes q_{ps} proxy signing queries. For each queries on a message m_i with warrant w_i such that $1 \leq i \leq q_{ps}$, $\mathscr{B}$ simulates the corresponding proxy signature as follows:

(a) If $ID_{1_i} \neq ID_\alpha$, $ID_{2_i} \neq ID_\alpha$ assume $H_0(ID_{1_i}) = c_{1_i}P$, $H_0(ID_{2_i}) = c_{2_i}P$; then $\mathscr{B}$ chooses two random numbers $r_{1_i}, r_{2_i} \in \mathbb{Z}_q^*$ and returns the proxy signature $\sigma_i = (\sigma_{M_{i_1}}, \sigma_{M_{i_2}}, \sigma_{M_{i_3}})$ such that $\sigma_{M_{i1}} = c_{1_i}aP + r_{1_i}H_1(w_i) + c_{2_i}aP + r_{2_i}H_2(w_i, m_i) + r_{2_i}H_2(m_i)$, $\sigma_{M_{i_2}} = (r_{1_i} + r_{2_i})P$ and $\sigma_{M_{i3}} = r_{2_i}P$ to $\mathscr{A}_{II}$. It is a correct simulation since

$$\begin{aligned}&e\left(\sigma_{M_{i1}}, P\right) = e\left(c_{1_i}aP + r_{1_i}H_1(w_i) + c_{2_i}aP\right. \\ &\left. + r_{2_i}H_2(w_i, m_i) + r_{2_i}H_1(w_i), P\right) = e\left(c_{1_i}P, aP\right) \\ &\cdot e\left(H_1(w_i), \left(r_{1_i} + r_{2_i}\right)P\right) e\left(c_{2_i}P, aP\right) \\ &\cdot e\left(H_2(m_i, w_i), r_{2_i}P\right) = e\left(H_0\left(ID_{1_i}\right), P_{pub}\right) \\ &\cdot e\left(H_1(w_i), \sigma_{M_{i_2}}\right) e\left(H_0\left(ID_{2_i}\right), P_{pub}\right) \\ &\cdot e\left(H_2(m_i, w_i), \sigma_{M_{i_3}}\right)\end{aligned} \tag{10}$$

(b) If $ID_{1_i} \neq ID_\alpha$, $ID_{2_i} = ID_\alpha$, assume $H_0(ID_{1_i}) = c_{1_i}P$, $H_0(ID_{2_i}) = c_\alpha P + bP$; then

(i) If $w_i \neq w_\beta|_{ID_\alpha \longrightarrow o}, m_i \neq m_\delta$ or $w_i \neq w_\beta|_{ID_\alpha \longrightarrow o}, m_i = m_\delta$, $\mathscr{B}$ terminates the simulation and reports failure.

(ii) If $w_i \neq w_\beta|_{ID_\alpha \longrightarrow p}$ and $m_i \neq m_\delta$, assume $H_1(w_i) = k_iP$ and $H_2(w_i, m_i) = u_iP + aP$; $\mathscr{B}$ simulates the proxy signature $\sigma_i = (\sigma_{M_{i_1}}, \sigma_{M_{i_2}}, \sigma_{M_{i_3}})$ by setting $\sigma_{M_{i_3}} = r_{2_i}P = v_iP - bP$, $\sigma_{M_{i_2}} = r_{1_i}P + v_iP - bP$ and $\sigma_{M_{i_1}} = (c_{1_i} + c_\alpha + v_i)P_{pub} + r_{1_i}H_2(w_i) + k_i(v_iP - bP) + u_i(v_iP - bP)$, where $v_i, r_{1_i} \in \mathbb{Z}_q$. It can be verified that it is a correct simulation since

$$\begin{aligned}&e\left(\sigma_{M_{i1}}, P\right) = e\left(\left(c_{1_i} + c_\alpha + v_i\right)P_{pub} + r_{1_i}H_2(w_i)\right. \\ &\left. + k_i(v_iP - bP) + u_i(v_iP - bP), P\right) = e\left(c_{1_i}P, aP\right) \\ &\cdot e\left(H_1(w_i), \left(r_{1_i} + r_{2_i}\right)P\right) e\left(abP + c_\alpha aP + v_iaP\right. \\ &\left. + u_iv_iP - u_ibP - abP, P\right) = e\left(c_{1_i}P, aP\right) \\ &\cdot e\left(H_1(w_i), \left(r_{1_i} + r_{2_i}\right)P\right) e\left(a\left(bP + c_\alpha P\right), P\right) \\ &\cdot e\left((v_i - b)\left(u_iP + aP\right), P\right) = e\left(H_0\left(ID_{1_i}\right), P_{pub}\right) \\ &\cdot e\left(H_1(w_i), \sigma_{M_{i_2}}\right) e\left(H_0\left(ID_{2_i}\right), P_{pub}\right) \\ &\cdot e\left(H_2(w_i, m_i), \sigma_{M_{i_3}}\right)\end{aligned} \tag{11}$$

(iii) If $w_i \neq w_\beta|_{ID_\alpha \longrightarrow p}, m_i = m_\delta$ or $w_i = w_\beta, m_i \neq m_\delta$, $\mathscr{B}$ performs the same as that in case (ii).

(iv) If $w_i = w_\beta$ and $m_i = m_\delta$, $\mathscr{B}$ terminates the simulation and reports failure.

(c) If $ID_{1_i} = ID_\alpha$, $ID_{2_i} \neq ID_\alpha$, assume $H_0(ID_{1_i}) = c_\alpha P + bP$, $H_0(ID_{2_i}) = c_{2_i}P$, then

(i) if $w_i \neq w_\beta|_{ID_\alpha \longrightarrow o}$ and $m_i \neq m_\delta$, assume $H_1(w_i) = k_iP - bP$ and $H_2(w_i, m_i) = u_iP + aP$. $\mathscr{B}$ chooses $l_i, r_{2_i} \in \mathbb{Z}_q^*$ and simulates the proxy signature $\sigma_i = (\sigma_{M_{i_1}}, \sigma_{M_{i_2}}, \sigma_{M_{i_3}})$ by setting $\sigma_{M_{i_3}} = r_{2_i}P$, $\sigma_{M_{i_2}} = v_iP - bP + r_{2_i}P$ and $\sigma_{M_{i_1}} = (c_\alpha + k_i + c_{2_i})P_{pub} + l_i(k_iP - bP) + r_{2_i}(k_iP - bP) + r_{2_i}(u_iP + aP)$. It is a correct simulation since

$$\begin{aligned}&e\left(\sigma_{M_{i1}}, P\right) = e\left(\left(c_\alpha + k_i + c_{2_i}\right)P_{pub} + l_i\left(k_iP - bP\right)\right. \\ &\left. + r_{2_i}\left(k_iP - bP\right) + r_{2_i}\left(u_iP + aP\right), P\right) = e\left(abP\right. \\ &\left. + ac_\alpha P + l_ik_iP - l_ibP + ak_iP - abp\right. \\ &\left. + r_{2_i}\left(k_iP - bP\right), P\right) e\left(c_{2_i}P_{pub}, P\right) \\ &\cdot e\left(r_{2_i}\left(u_iP + aP\right), P\right) = e\left(a\left(c_\alpha P + bP\right), P\right)\end{aligned}$$

$$\cdot e\left(\left(l_i + a + r_{2_i}\right)\left(k_i P - bP\right), P\right) e\left(c_{2_i} P, aP\right) e\left(u_i P + aP, r_{2_i} P\right) = e\left(H_0\left(ID_{1_i}\right), P_{pub}\right) \cdot e\left(H_1\left(w_i\right), \sigma_{M_{i_2}}\right) e\left(H_0\left(ID_{2_i}\right), P_{pub}\right) \cdot e\left(H_2\left(w_i, m_i\right), \sigma_{M_{i_3}}\right) \tag{12}$$

(ii) If $w_i \neq w_\beta|_{ID_\alpha \longrightarrow p}, m_i \neq m_\delta$ or $w_i \neq w_\beta|_{ID_\alpha \longrightarrow p}, m_i = m_\delta$, $\mathscr{B}$ terminates the simulation and reports failure.

(iii) If $w_i \neq w_\beta|_{ID_\alpha \longrightarrow 0}$ and $m_i = m_\delta$, assume $H_1(w_i) = k_i P - bP$ and $H_2(w_i, m_\beta) = u_i P + aP$; $\mathscr{B}$ performs the same as that in case (i).

(iv) If $w_i = w_\beta$ and $m_i \neq m_\delta$, assume $H_1(w_\beta) = k_i P$ and $H_2(w_\beta, m_i) = u_i P + aP$; $\mathscr{B}$ chooses $v_i, r_{1_i} \in \mathbb{Z}_q^*$ and simulates the proxy signature $\sigma_i = (\sigma_{M_{i_1}}, \sigma_{M_{i_2}}, \sigma_{M_{i_3}})$ by setting $\sigma_{M_{i_3}} = v_i P - bP$, $\sigma_{M_{i_2}} = v_i P - bP + r_{1_i} P$, and $\sigma_{M_{i_1}} = (c_\alpha + c_{2_i} + v_i) P_{pub} + r_{1_i} k_i P + k_i (v_i P - bP) + u_i (v_i P - bP)$. It is a correct simulation since

$$\begin{aligned} e\left(\sigma_{M_{i1}}, P\right) &= e\left(\left(c_\alpha + c_{2_i} + v_i\right) P_{pub} + r_{1_i} k_i P + k_i \left(v_i P - bP\right) + u_i \left(v_i P - bP\right), P\right) \\ &= e\left(\left(c_\alpha + b\right) aP + c_{2_i} aP + r_{1_i} k_i P + k_i \left(v_i P - bP\right) + \left(v_i - b\right) aP + u_i \left(v_i P - bP\right), P\right) = e\left(c_\alpha P + bP, aP\right) e\left(k_i P, r_{1_i} P + v_i P - bP\right) e\left(c_{2_i} P, aP\right) \\ &\cdot e\left(u_i P, v_i P - bP\right) = e\left(H_0\left(ID_{1_i}\right), P_{pub}\right) \cdot e\left(H_1\left(w_i\right), \sigma_{M_{i_2}}\right) e\left(H_0\left(ID_{2_i}\right), P_{pub}\right) \cdot e\left(H_2\left(w_i, m_i\right), \sigma_{M_{i_3}}\right) \end{aligned} \tag{13}$$

(v) If $w_i = w_\beta$ and $m_i = m_\delta$, $\mathscr{B}$ terminates the simulation and reports failure.

(d) If $ID_{1_i} = ID_\alpha$, $ID_{2_i} = ID_\alpha$, $\mathscr{B}$ terminates the simulation and reports failure.

(6) **Forgery**: assume $\mathscr{A}_{II}$ outputs a valid proxy signature $\sigma^* = (\sigma^*_{M_1}, \sigma^*_{M_2}, \sigma^*_{M_3})$ on message M^* under a warrant W^* with the proxy signer's identity ID_A and the proxy signer's identity ID_B. Besides,

(i) ID_A has not been queried in the key extraction queries,

(ii) (ID_A, W^*) has not been queried in the delegation queries,

(iii) (ID_A, ID_B, W^*, M^*) has not been queried in the proxy signing queries,

If $H_0(ID_A) \neq bP + c_\alpha P$ or $H_1(W^*) \neq k_\beta P$ or $H_2(W^*, M^*) \neq u_\delta P$, $\mathscr{B}$ will abort. Otherwise, given the forged proxy signature $\sigma^* = (\sigma^*_{M_1}, \sigma^*_{M_2}, \sigma^*_{M_3})$. $\mathscr{B}$ can solve the CDH problem

$$abP = \sigma^*_{M_1} - c_\alpha aP - k_\beta \sigma^*_{M_2} - c_{2_i} aP - u_\delta \sigma^*_{M_3} \tag{14}$$

when $H_0(ID_A) = bP + c_\alpha P$, $H_1(ID_B) = k_\beta P$, and $H_2(W^*, M^*) = u_\delta P$.

Next, we analyze the success probability of $\mathscr{B}$; $\mathscr{B}$ will not abort if the following conditions hold:

(i) $ID_A = ID_\alpha$.

(ii) $W^* = w_\beta$.

(iii) $M^* = m_\delta$.

Therefore, if $\mathscr{A}_{II}$ has a nonnegligible probability ϵ in breaking the proposed ID-based proxy signature scheme, then the success probability of $\mathscr{B}$ in solving CDH problem is

$$Succ^{CDH}_{\mathscr{B},\mathbb{G}_1} \geq \frac{\epsilon}{\left(q_{H_0} + q_k + q_{os's} + 2q_{ps}\right)\left(q_{H_1} + q_{os's} + q_{ps}\right)\left(q_{H_2} + q_{ps}\right)}. \tag{15}$$

which is nonnegligible. Thus, we reach a contradiction. □

Theorem 7. *The revised ID-based proxy signature scheme is secure against the $\mathscr{A}_{III}$ chosen message and identity attack if the CDH assumption holds.*

Proof. The security is similar to that in Theorem 6. Thus, we just describe it briefly.

(1) **Setup**, **Hash queries,** and **Key extract** queries are the same as those in the security proof against a malicious proxy signer.

(2) **Proxy signer's standard signing queries** and **Proxy signing queries** are similar to the **Original signer's stand signing queries** and **Proxy signing queries** in the security for Theorem 6.

Through simulation, it can be reduced that if there exists a malicious original signer that can break the proposed scheme with a nonnegligible probability ϵ, then we can build another probabilistic polynomial time algorithm $\mathscr{B}$ that can solve the CDH problem with a nonnegligible probability $Succ^{CDH}_{\mathscr{B},\mathbb{G}_1}$ such that

$$Succ^{CDH}_{\mathscr{B},\mathbb{G}_1} \geq \frac{\epsilon}{\left(q_{H_0} + q_k + q_{ps's} + 2q_{ps}\right)\left(q_{H_1} + q_{ps's} + q_{ps}\right)\left(q_{H_2} + q_{ps}\right)} \tag{16}$$

where $q_{ps's}$ refers to the number of proxy signer's standard signing queries. Thus, we reach a contradiction. □

Table 1: Comparison regarding the computational costs.

Schemes	ProSign	ProVer
Wu et al.'s scheme [13]	$2 \cdot A_{\mathbb{G}_1} + 2 \cdot M_{\mathbb{G}_1} + 1 \cdot T_H$	$5 \cdot P + 4 \cdot T_H$
Our scheme	$3 \cdot A_{\mathbb{G}_1} + 4 \cdot M_{\mathbb{G}_1} + 2 \cdot T_H$	$5 \cdot P + 4 \cdot T_H$

6.1. Efficiency Analysis. We analyze the efficiency of the revised proxy signature scheme and compare it with the original scheme. The detail computation costs are presented in Table 1. As have been noticed, some algorithms in the revised scheme remains unchanged; thus, we only concern those algorithms that are different in our and the original schemes. Let $M_{\mathbb{G}_1}$, $A_{\mathbb{G}_1}$ denote the multiplication add addition calculations in $\mathbb{G}_1$, T_H denote the calculation of hash function (either H_0, H_1, or H_2), and let P denote the calculation of paring. We can see that our revised proxy signature scheme involves only one addition, two multiplication, and one hash operation in the proxy signing algorithm. As for the expensive paring operations needed in the proxy verification parts, the numbers are exactly the same.

7. Conclusion

In this paper, we introduced a practical attack which has not been considered by some existing proxy signature schemes. In particular, we took an identity-based proxy signature scheme to describe how this attack works. We also presented an enhanced security model that can capture this attack. Our model has considered different types of potential adversaries against an identity-based proxy signature scheme and allowed the adversary to query the individual signatures of both the original signer and the proxy signer. The proposed new scheme inherits the good features of the original scheme and at the same time can effectively prevent the attack. The proposed method can also be applied in other proxy signature schemes [18–21] to ensure an improved security.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors gratefully thank Xinyi Huang and Yong Yu for discussions on this work.

References

[1] W. Ren, R. Liu, M. Lei, and K.-K. R. Choo, "SeGoAC: A tree-based model for self-defined, proxy-enabled and group-oriented access control in mobile cloud computing," *Computer Standards & Interfaces*, vol. 54, pp. 29–35, 2017.

[2] Y. Yang, H. Zhu, H. Lu, J. Weng, Y. Zhang, and K.-K. R. Choo, "Cloud based data sharing with fine-grained proxy re-encryption," *Pervasive and Mobile Computing*, vol. 28, pp. 122–134, 2016.

[3] C. Calvelli and V. Varadharajan, "An analysis of some delegation protocols for distributed systems," in *Proceedings of the [1992] The Computer Security Foundations Workshop V*, pp. 92–110, Franconia, NH, USA.

[4] B. Neuman, "Proxy-based authorization and accounting for distributed systems," in *Proceedings of the [1993]. The 13th International Conference on Distributed Computing Systems*, pp. 283–291, Pittsburgh, PA, USA.

[5] B. Lee, H. Kim, and K. Kim, "Strong proxy signature and its applications," in *Proceedings of the SCIS*, vol. 1, pp. 603–608.

[6] I. Foster, C. Kesselman, G. Tsudik, and S. Tuecke, "Security architecture for computational grids," in *Proceedings of the 1998 5th ACM Conference on Computer and Communications Security, CCS-5*, pp. 83–92, November 1998.

[7] X. Jia, D. He, Q. Liu, and K. R. Choo, "An efficient provably-secure certificateless signature scheme for Internet-of-Things deployment," *Ad Hoc Networks*, vol. 71, pp. 78–87, 2018.

[8] A. Castiglione, K. Raymond Choo, M. Nappi, and S. Ricciardi, "Context Aware Ubiquitous Biometrics in Edge of Military Things," *IEEE Cloud Computing*, vol. 4, no. 6, pp. 16–20, 2017.

[9] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures for delegating signing operation," in *Proceedings of the 3rd ACM Conference on Computer and Communications Security*, pp. 48–56, ACM Press, March 1996.

[10] S. Kim, S. Park, and D. Won, "Proxy signatures, revisited," in *Information and Communications Security*, vol. 1334 of *Lecture Notes in Computer Science*, pp. 223–232, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997.

[11] Z. Fangguo, R. Safavi-Naini, and L. Chih-Yin, "New proxy signature, proxy blind signature and proxy ring signature schemes from bilinear pairing," *CiteSeer*, 2003.

[12] G. Fuchsbauer and D. Pointcheval, "Anonymous Proxy Signatures," in *Security and Cryptography for Networks*, vol. 5229 of *Lecture Notes in Computer Science*, pp. 201–217, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[13] B. Xiao, L. T. Yang, J. Ma, C. Muller-Schloer, and Y. Hua, "Identity-based proxy signature from pairings," in *Proceedings of the Autonomic and Trusted Computing, 4th International Conference, ATC 2007*, vol. 4610, Springer Berlin Heidelberg, Hong Kong, China, July 2007.

[14] K. Zhang, "Threshold proxy signature schemes," in *Proceedings of the Information Security, First International Workshop, ISW '97*, pp. 282–290, Tatsunokuchi, Japan, September 1997.

[15] B. Lee, H. Kim, and K. Kim, "Secure mobile agent using strong non-designated proxy signature," in *Information Security and Privacy: Proceedings of the 6th Australasian Conference (ACISP '01), Sydney, Australia, July 11–13, 2001*, vol. 2119 of *Lecture Notes in Computer Science*, pp. 474–486, Springer, Berlin, Germany, 2001.

[16] G. Wang, "Designated-Verifier Proxy Signature Schemes," in *Security and Privacy in the Age of Ubiquitous Computing*, vol. 181 of *IFIP Advances in Information and Communication Technology*, pp. 409–423, Springer US, Boston, MA, 2005.

[17] W. Liu, G. Yang, Y. Mu, and J. Wei, "k-time proxy signature: formal definition and efficient construction," in *Provable security*, vol. 8209 of *Lecture Notes in Computer Science*, pp. 154–164, Springer, Heidelberg, 2013.

[18] X. Huang, W. Susilo, Y. Mu, and W. Wu, "Proxy signature without random oracles," in *Mobile Ad-Hoc and Sensor Networks*, vol. 4325, pp. 473–484, Springer, Berlin, Germany, 2006.

[19] L. Jin, K. Kwangjo, Z. Fangguo, and C. Xiaofeng, "Aggregate proxy signature and verifiably encrypted proxy signature," in *Proceedings of the Provable Security, First International Conference, ProvSec 2007*, pp. 208–217, Wollongong, Australia, 2007.

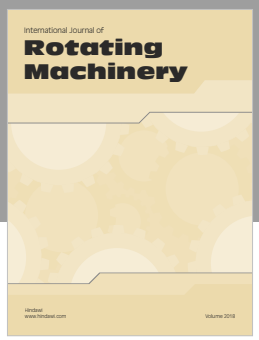
[20] Y. Sun, C. X. Xu, Y. Yu, and Y. Mu, "Strongly unforgeable proxy signature scheme secure in the standard model," *The Journal of Systems and Software*, vol. 84, no. 9, pp. 1471–1479, 2011.

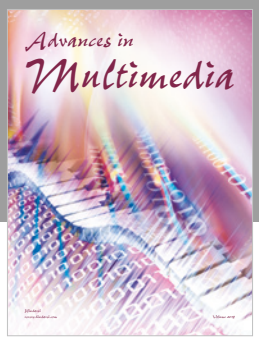
[21] W. Liu, Y. Mu, and G. Yang, "Attribute-Based Signing Right Delegation," in *Network and System Security*, vol. 8792 of *Lecture Notes in Computer Science*, pp. 323–334, Springer International Publishing, Cham, 2014.

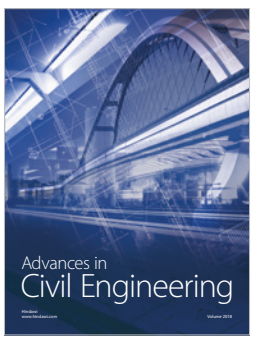
[22] A. Boldyreva, A. Palacio, and B. Warinschi, "Secure proxy signature schemes for delegation of signing rights," *Journal of Cryptology*, vol. 25, no. 1, pp. 57–115, 2012.

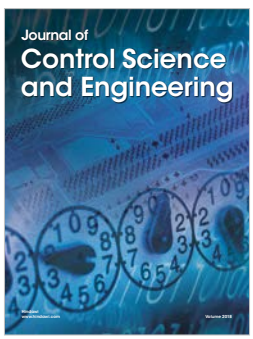
[23] T. Malkin, S. Obana, and M. Yung, "The hierarchy of key evolving signatures and a characterization of proxy signatures," in *Advances in Cryptology—EUROCRYPT 2004*, vol. 3027 of *Lecture Notes in Computer Science*, pp. 306–322, Springer, Berlin, Germany, 2004.

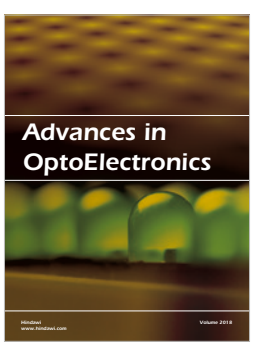
[24] J. C. N. Schuldt, K. Matsuura, and K. G. Paterson, "Proxy signature secure against key exposure," in *Public Key Cryptography—PKC 2008: 11th International Workshop on Practice and Theory in Public-Key Cryptography, Barcelona, Spain, March 9-12, 2008. Proceedings*, vol. 4939 of *Lecture Notes in Computer Science*, pp. 141–161, Springer, Berlin, Germany, 2008.

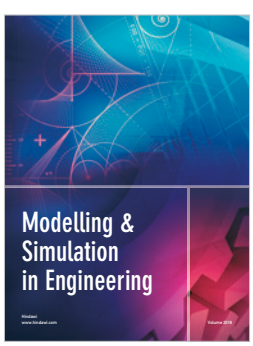


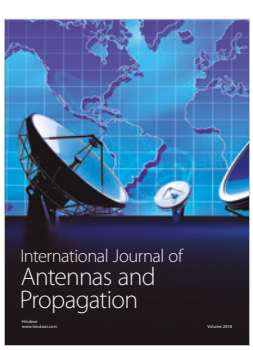


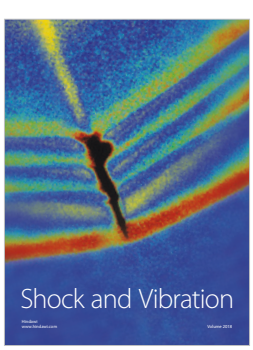


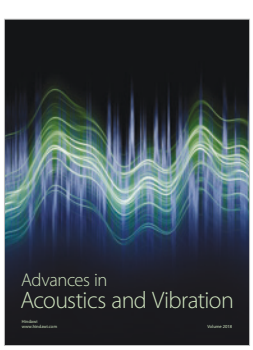












Hindawi

Submit your manuscripts at
www.hindawi.com