Cryptanalysis and improvement of a certificateless multi-proxy signature scheme

Miaomiao Tian*, Wei Yang, and Liusheng Huang

School of Computer Science and Technology, University of Science and Technology of China, Hefei, 230026, China
Suzhou Institute for Advanced Study, University of Science and Technology of China, Suzhou, 215123, China

Abstract. Multi-proxy signature allows an original signer authorizing a proxy group as his proxy agent and only the cooperation of all proxy signers in the group can create a proxy signature on behalf of the original signer. Recently, Jin and Wen defined a formal model of certificateless multi-proxy signature and proposed a concrete scheme. They claimed that their scheme is provably secure in their security model. Unfortunately, by giving concrete attacks, we show that Jin-Wen's certificateless multi-proxy signature scheme is not secure according to their security model. Possible improvements of their scheme are also suggested to prevent these attacks.

Keywords: Certificateless cryptography; Multi-proxy signature; Bilinear pairing; Cryptanalysis

1 Introduction

In a traditional public key infrastructure, a digital certificate binding a user with his public key needs to be produced by a Certification Authority (CA). It brings the certificate management problem since such a system requires a large amount of computing and storage cost to manage certificates. To resolve this problem, Shamir [1] first introduced the concept of identity-based (ID-based) public key cryptography in 1984. In this setting, a user's public key is derived from his identity, e.g., his email address, and his secret key is generated by a trusted third party called the private key generator (PKG). However, ID-based public key cryptography inevitably suffers from the key escrow problem, namely the PKG knows all the user's secret keys.

In 2003, Al-Riyami and Paterson [2] proposed the notion of certificateless public key cryptography to eliminate the key escrow problem in ID-based cryptography and the use of certificates in traditional public key cryptography simultaneously. In such a cryptosystem, the PKG only generates a partial private key for a user. The user then combines his partial private key with some secret value chosen by the user himself to generate his full private key. Therefore,

* Corresponding author. E-mail: miaotian@mail.ustc.edu.cn (M. Tian).

certificateless public key cryptography is more interesting as it combines the benefit of the traditional public key cryptography and the ID-based public key cryptography. After the Al-Riyami and Paterson's seminal work [2], a number of papers have been published in this area including several certificateless proxy and multi-proxy signature proposals, e.g., [3–9].

The concept of proxy signature was introduced by Mambo *et al.* [10] in 1996. Proxy signatures allow an original signer delegating a proxy signer to sign messages on its behalf. Proxy signature schemes have found numerous practical applications such as grid computing [11], mobile agent systems [12, 13] and cloud applications [14]. In 2000, Hwang and Shi [15] first presented a new type of proxy signature called multi-proxy signature. In a multi-proxy signature scheme, an original signer could authorize a proxy group as his proxy agent and only the cooperation of all proxy signers in the group can create a proxy signature on behalf of the original signer. Up to now, a number of multi-proxy signature schemes have been proposed [9, 15–17]. Recently, Jin and Wen [9] presented the first certificateless multi-proxy signature scheme along with a security model. They claimed that their scheme is secure according to their security model. However, in this paper, by giving concrete attacks, we show that Jin-Wen's scheme is not secure in their security model. We also present some suggestions on the possible improvements.

The rest of this paper is organized as follows. The fundamental background knowledge of bilinear pairing and Jin-Wen's formal model of certificateless multi-proxy signature are given in Section 2. In Section 3, we review Jin-Wen's certificateless multi-proxy signature scheme. In section 4, we present two concert attacks against Jin-Wen's scheme as well as the corresponding improvements to prevent these attacks. Finally, we conclude the paper in Section 5.

2 Preliminaries

In this paper, we assume that there are $n \in \mathbb{Z}_{>0}$ proxy signers in the proxy group. For a positive integer l, $[l]$ denotes $\{1, \ldots, l\}$.

2.1 Bilinear pairing

Let $\mathbb{G}$ and $\mathbb{G}_T$ be two cyclic groups with the same prime order q, and let P be a generator of $\mathbb{G}$. A map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is called a bilinear map if it satisfies the following three properties.

1. Bilinearity: For all $a, b \in \mathbb{Z}$ and $Q \in \mathbb{G}$, we have $e(aP, bQ) = e(P, Q)^{ab}$.
2. Non-degeneracy: Let $1_{\mathbb{G}_T}$ denote the identity element of group $\mathbb{G}_T$, then we have $e(P, P) \neq 1_{\mathbb{G}_T}$.
3. Computable: There exists an efficient algorithm to compute $e(aP, Q)$ for any $a \in \mathbb{Z}$ and $Q \in \mathbb{G}$.

2.2 Certificateless multi-proxy signature scheme and its security model

In this subsection, we present the syntax and security model of certificateless multi-proxy signature specified in [9].

Certificateless multi-proxy signature scheme A certificateless multi-proxy signature scheme consists of the following eight probabilistic polynomial-time algorithms:

- **Setup:** On input a security parameter k, the PKG generates a master secret key MSK and the public parameters PP. The PKG publishes the public parameters PP and keeps the master key MSK secret.
- **Partial-Private-Key-Generate:** On input the master secret key MSK and a user's identity ID, the PKG generates a partial secret key psk_{ID} for the user.
- **User-Key-Generate:** Given the master secret key MSK and a user's identity ID, the user selects a random secret value x_{ID} and then constructs his full public/secret key pair (pk_{ID}, sk_{ID}).
- **Sign:** On input a message m, user identity ID, the secret key sk_{ID} of the signer ID, this algorithm outputs a signature σ on m.
- **Verify:** On input a signature σ, a message m, an identity ID and the corresponding public key pk_{ID}, it returns 1 if σ is a valid signature on m for the user, and returns 0 otherwise.
- **Proxy-Key-Generate:** It is an interactive algorithm between the original signer ID_0 and all proxy signers $ID_i (i \in [n])$. Let w denote the delegation warrant which records the delegation police and the identities of the original signer and all proxy singers. On input the warrant w, all identities $ID_0, ID_1, \ldots, ID_n$ and private keys $sk_0, sk_1, \ldots, sk_n$, it outputs a multi-proxy secret key $mpsk_i$ for proxy signer ID_i, where $i \in [n]$.
- **MP-Sign:** On input a message m satisfying w, all proxy private keys $mpsk_1, \ldots, mpsk_n$, it returns a multi-proxy signature σ_{mp} on behalf of the original signer ID_0.
- **MP-Verify:** On input the identities $ID_0, ID_1, \ldots, ID_n$, the public keys $pk_{ID_0}, pk_{ID_1}, \ldots, pk_{ID_n}$, a multi-proxy signature σ_{mp}, a message m and a warrant w, it outputs 1 if σ_{mp} is a valid multi-proxy signature, and outputs 0 otherwise.

Security model Based on the security models in [4], [5] and [16], Jin and Wen [9] presented a security model for certificateless multi-proxy signature where two types of adversaries were considered. The Type I adversary $\mathcal{A}_I$ does not know the master secret key MSK of the system. However, he is able to replace any user's public key with some value chosen by himself. The Type II adversary $\mathcal{A}_{II}$, called malicious-but-passive PKG, can generate the system parameters PP and the master secret key MSK maliciously at the very beginning of the setup stage of the system while he cannot replace any public key. Here, we recall Jin-Wen's security model defined via the following two games.

Game 1: This game is played between a challenger $\mathcal{C}$ and a polynomial time Type I adversary $\mathcal{A}_I$.

- **Setup.** The challenger $\mathcal{C}$ runs the Setup algorithm to obtain a master secret key MSK and the public parameters PP. Then $\mathcal{C}$ sends only PP to the adversary $\mathcal{A}_I$.
- **Public-Key-Inquiry:** $\mathcal{A}_I$ adaptively delivers queries on user identity ID, then $\mathcal{C}$ returns the public key pk_{ID} of the user to $\mathcal{A}_I$.
- **Public-Key-Replacement:** $\mathcal{A}_I$ replaces the public key pk_{ID} of the user ID with a different value pk^*_{ID} of his choice. $\mathcal{C}$ will record this replacement.
- **Partial-Private-Key-Extraction:** $\mathcal{A}_I$ submits a user's identity ID, $\mathcal{C}$ returns the partial private key psk_{ID} of the user to $\mathcal{A}_I$.
- **Secret-Value-Extraction:** $\mathcal{A}_I$ delivers queries on the user ID, $\mathcal{C}$ returns the symbol $\perp$ if the public key of user ID has been replaced; otherwise, he returns the secret value x_{ID} of the user to $\mathcal{A}_I$.
- **Signing-Query:** $\mathcal{A}_I$ chooses an identity ID and a message m. $\mathcal{C}$ returns the symbol $\perp$ if the public key of this user has been replaced; otherwise, he runs the Sign algorithm and returns a signature σ to $\mathcal{A}_I$.
- **Delegation-Query:** There are two types of queries.
  - Q_1. As the role of the original signer ID_0, $\mathcal{A}_I$ submits a warrant w of his choice. $\mathcal{C}$ runs the Proxy-Key-Generate algorithm as the role of a proxy signer $ID_i (i \in [n])$ in the proxy group and adds $(w, mpsk_i)$ to a list Warrp. We stress that $\mathcal{A}_I$ does not have access to the elements of Warrp.
  - Q_2. As the role of a proxy signer $ID_i (i \in [n])$ in the proxy group, $\mathcal{A}_I$ submits a warrant w of his choice. $\mathcal{C}$ runs the Proxy-Key-Generate algorithm as the role of the original signer ID_0, sends a corresponding multi-proxy signature secret key $mpsk_i$ to $\mathcal{A}_I$ and adds w to a list Warro.
- **Multi-Proxy-Signing-Query:** $\mathcal{A}_I$ delivers a multi-proxy signature query on (w, m). $\mathcal{C}$ checks if m satisfies w, the public keys of all proxy users and the original signer have not been replaced and there is a $mpsk_i (i \in [n])$ such that $(w, mpsk_i) \in Warrp$. If all the verifications pass, $\mathcal{C}$ runs the MP-Sign algorithm and returns a multi-proxy signature σ_{mp} to $\mathcal{A}_I$; otherwise, he returns $\perp$.
- **Forgery.** Finally, $\mathcal{A}_I$ outputs a forgery and wins the game if any of the following events occurs:
  - E_1*:* $\mathcal{A}_I$ forges a valid signature σ^* on a message m^* with respect to identity ID^*, where σ^* is not an output of the Signing-Query and ID^* has not been submitted to both the Partial-Private-Key-Extraction oracle and either the Secret-Value-Extraction oracle or the Public-Key-Replacement oracle.
  - E_2*:* $\mathcal{A}_I$ forges a valid multi-proxy signature σ^*_{mp} on a message m^* under the warrant w^*, where σ^*_{mp} is not an output of the Multi-Proxy-Signing-Query.
  - E_3*:* $\mathcal{A}_I$ forges a valid multi-proxy signature σ^*_{mp} on a message m^* under the warrant w^*, where $w* \notin Warro$.

Game 2: This game is played between a challenger $\mathcal{C}$ and a Type II adversary $\mathcal{A}_{II}$.

- **Setup.** The adversary $\mathcal{A}_{II}$ runs the Setup algorithm to obtain a master secret key MSK and the public parameters PP. Then $\mathcal{A}_{II}$ sends both PP and MSK to $\mathcal{C}$. Note that the public parameters are chosen by $\mathcal{A}_{II}$.
- **Queries.** The adversary $\mathcal{A}_{II}$ may adaptively make a polynomially bounded number of the following queries: Public-Key-Inquiry, Secret-Value-Extraction, Signing-Query, Delegation-Query and Multi-Proxy-Signing-Query, which are similar as those in Game 1. Note that $\mathcal{A}_{II}$ cannot replace any public key in this Game.
- **Forgery.** Finally, $\mathcal{A}_{II}$ outputs a forgery and wins the game if any of the following events occurs:
  - E_1*:* $\mathcal{A}_{II}$ forges a valid signature σ^* on a message m^* with respect to identity ID^*, where σ^* is not an output of the Signing-Query and ID^* has not been submitted to the Secret-Value-Extraction oracle.
  - E_2*:* $\mathcal{A}_{II}$ forges a valid multi-proxy signature σ^*_{mp} on a message m^* under the warrant w^*, where σ^*_{mp} is not an output of the Multi-Proxy-Signing-Query.
  - E_3*:* $\mathcal{A}_{II}$ forges a valid multi-proxy signature σ^*_{mp} on a message m^* under the warrant w^*, where $w* \notin Warro$.

Definition 1. A certificateless multi-proxy signature scheme is said to be existentially unforgeable against adaptively chosen warrant attacks and chosen message and identity attacks if there exists no polynomial time adversary who wins any of the above games with non-negligible probability.

3 Review of Jin-Wen's certificateless multi-proxy signature scheme

In this section, we review Jin-Wen's certificateless multi-proxy signature scheme [9], which is specified as follows.

Setup: Given a security parameter k, the PKG chooses two groups $\mathbb{G}$ and $\mathbb{G}_T$ with same prime order q, a bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$, the master secret key $s \in \mathbb{Z}_q^*$ and a generator P of $\mathbb{G}$. It also chooses six different cryptographic hash functions $H_1, H_2, H_3 : \{0,1\}^* \to \mathbb{G}$ and $H_4, H_5, H_6 : \{0,1\}^* \to \mathbb{Z}_q^*$. The PKG publishes the public parameters $PP = (\mathbb{G}, \mathbb{G}_T, e, P, Q, H_1, H_2, H_3, H_4, H_5, H_6)$, where $Q = sP$ is the master public key of the system.

Partial-Private-Key-Extract: Given a user's identity $ID_i \in \{0,1\}^*$, the PKG generates the partial private key $D_i = sH_1(ID_i)$ for the user.

User-Key-Generate: The user ID_i selects a random number $x_i \in \mathbb{Z}_q^*$ and sets his private key as $sk_i = (x_i, D_i)$. His public key is $pk_i = x_iP$.

Sign: On input a message $m \in \{0,1\}^*$, the signer ID_i with private key sk_i does the following steps:

1. Choose $r \in \mathbb{Z}_q^*$ uniformly at random and compute $R = rP$.

2. Compute $W = H_2(PP)$, $T = H_3(Q)$ and $h = H_4(PP||m||ID_i||pk_i||R)$.
3. Compute $V = hD_i + x_iW + rT$.

The signature on m is $\sigma = (R, V)$.

Verify: To verify a signature $\sigma = (R, V)$ on message m with respect to identity ID_i and public key pk_i, the verifier checks whether $e(V, P) = e(hH_1(ID_i), Q)e(W, pk_i)e(T, R)$, where $W = H_2(PP)$, $T = H_3(Q)$ and $h = H_4(PP||m||ID_i||pk_i||R)$. If the verification passes, output 1; otherwise, output 0.

Proxy-Key-Generate:

1. **Delegation-generation:** To delegate the signing capability, the original signer ID_0 signs on the warrant w which specifies some proxy details, such as the identities of the original signer and the proxy signers, the type of messages delegated and the period of delegation.
   - Choose a random number $r_0 \in \mathbb{Z}_q^*$ and compute $R_0 = r_0P$.
   - Compute $W = H_2(PP)$, $T = H_3(Q)$, $h_0 = H_5(PP||w||ID_0||pk_0||R_0)$ and $V_0 = h_0D_0 + x_0W + r_0T$.
   - Send (w, R_0, V_0) to each proxy signer $ID_i (i \in [n])$.
2. **Delegation-verification:** Each proxy signer $ID_i (i \in [n])$ confirms (w, R_0, V_0) by checking $e(V_0, P) = e(h_0H_1(ID_0), Q)e(W, pk_0)e(T, R_0)$. ID_i accepts the delegation if the equation holds; otherwise, he requests a valid one from ID_0 or terminates the protocol.
3. **Proxy-secret-key-generation:** If the proxy signer $ID_i (i \in [n])$ confirms the delegation, he sets $mpsk_i = (sk_i, R_0, V_0)$ as his multi-proxy secret key.

MP-Sign: On input a message $m \in \{0, 1\}^*$ satisfying the warrant w and the identity ID_0 of the original signer, the proxy signer $ID_i (i \in [n])$ with multi-proxy secret key $mpsk_i$ performs as follows:

1. Choose a random number $r_i \in \mathbb{Z}_q^*$ and compute $R_i = r_iP$.
2. Compute $V_i = h_iD_i + x_iW + r_iT$, where $W = H_2(PP)$, $T = H_3(Q)$ and $h_i = H_6(PP||m||ID_i||pk_i||R_i)$.
3. Send his partial proxy signature (w, R_0, V_0, R_i, V_i) to the clerk of the proxy group.

After receiving (w, R_0, V_0, R_i, V_i) from ID_i, the clerk checks the following equations:

$$e(V_0, P) = e(h_0H_1(ID_0), Q)e(W, pk_0)e(T, R_0) \tag{1}$$

and

$$e(V_i, P) = e(h_iH_1(ID_i), Q)e(W, pk_i)e(T, R_i) \tag{2}$$

where $h_0 = H_5(PP||w||ID_0||pk_0||R_0)$ and $h_i = H_6(PP||m||ID_i||pk_i||R_i)$.

In the case that all partial multi-proxy signatures are valid, the multi-proxy signature on the message m under the warrant w is $\sigma_{mp} = (w, R_{mp}, V_{mp})$, where $R_{mp} = (R_0, R_1, \ldots, R_n)$ and $V_{mp} = \sum_{i=0}^{n} V_i$.

MP-Verify: To verify the multi-proxy signature $\sigma_{mp} = (w, R_{mp}, V_{mp})$ on the message m under the warrant w, one checks whether

$$e(V_{mp}, P) = e(\sum_{i=0}^{n} h_i H_1(ID_i), Q) e(W, \sum_{i=0}^{n} pk_i) e(T, \sum_{i=0}^{n} R_i) \tag{3}$$

holds or not, where $W = H_2(PP)$, $T = H_3(Q)$, $h_0 = H_5(PP||w||ID_0||pk_0||R_0)$ and $h_i = H_6(PP||m||ID_i||pk_i||R_i)(i \in [n])$. If it holds, output 1; otherwise, output 0.

4 Analysis and improvement of Jin-Wen's scheme

Jin and Wen [9] claimed that their scheme is provably secure in the above model. In this section, we disprove their claim by giving two concrete attacks. Concretely, there exists polynomial time adversary $\mathcal{A}$ who can always win Game 1 or Game 2. Then we propose an improved scheme to prevent these attacks.

4.1 Attack I

1. In the **Setup** phase, adversary $\mathcal{A}$ obtains the system parameters PP from the challenger $\mathcal{C}$ (in Game 1) or himself (in Game 2).
2. In the **Public-Key-Inquiry**, **Partial-Private-Key-Extraction** and **Secret-Value-Extraction** phases, $\mathcal{A}$ issues public key requests, partial private key requests (such requests can be removed in Game 2) and secret value requests with all the user's identities $ID_i(i \in \{0, 1, \ldots, n\})$, respectively. Then $\mathcal{A}$ is given all the participant's public keys $pk_i(i \in \{0, 1, \ldots, n\})$, partial private keys $D_i(i \in \{0, 1, \ldots, n\})$ and secret values $x_i(i \in \{0, 1, \ldots, n\})$. Note that $\mathcal{A}$ gets all the user's secrete keys $sk_i = (x_i, D_i)(i \in \{0, 1, \ldots, n\})$ after such queries.
3. In other phases, adversary $\mathcal{A}$ needn't issue any query.

The adversary $\mathcal{A}$ does the same operations as **Proxy-Key-Generate** and **MP-Sign** described in Section 3. Then he can get a valid multi-proxy signature σ_{mp} on a message m under the warrant w. Observe that, in this case, the events E_2 and E_3 both occur no matter in which Games. In other words, the polynomial time adversary $\mathcal{A}$ wins Game 1 or Game 2 with unit probability. Therefore, Jin-Wen's certificateless multi-proxy signature scheme is not secure.

Although this is an ordinary attack, it's proper according to their security model. The reason is Jin-Wen's security model is flawed. Actually, for a valid multi-proxy signature, it should also restrict that $\mathcal{A}$ cannot issue both the Partial-Private-Key-Extraction query and either the Secret-Value-Extraction query or the Public-Key-Replacement query on the original signer in Game 1 or the Secret-Value-Extraction query on the original signer in Game 2. In the next subsection, however, we will show that their scheme is still insecure even if $\mathcal{A}$ does not know the full secret key of the original signer.

4.2 Attack II

In this subsection, we assume that the security model has been renovated. Now, we illustrate how a polynomial time adversary $\mathcal{A}$ successfully attacks Jin-Wen's scheme [9].

1. In the **Setup** phase, adversary $\mathcal{A}$ obtains the system parameters PP from the challenger $\mathcal{C}$ (in Game 1) or himself (in Game 2).
2. In the **Queries** phrase, $\mathcal{A}$ makes public key request and partial private key request (this request also can be omitted in Game 2) on a user $ID_i (i \in \{0, 1, \ldots, n\})$. Then, $\mathcal{A}$ makes Signing-Query on a message m with respect to ID_i. The challenger $\mathcal{C}$ returns (pk_i, D_i) and (R_i, V_i) to $\mathcal{A}$ such that $e(V_i, P) = e(h_i H_1(ID_i), Q)e(W, pk_i)e(T, R_i)$, where $W = H_2(PP)$, $T = H_3(Q)$ and $h_i = H_4(PP||m||ID_i||pk_i||R_i)$.
3. For any message m^*, $\mathcal{A}$ makes H_4-Query on $(PP||m^*||ID_i||pk_i||R_i)$. Afterwards, $\mathcal{A}$ obtains $h_i^* = H_4(PP||m^*||ID_i||pk_i||R_i)$.

Without loss of generality, we assume that $a = h_i^* - h_i$, where $a \in \{2 - q, \ldots, q - 2\}$. $\mathcal{A}$ sets $V_i^* = V_i + aD_i$ and $R_i^* = R_i$. Then $\sigma^* = (R_i^*, V_i^*)$ is a valid signature of m^* since

$$\begin{aligned} e(V_i^*, P) &= e(V_i, P)e(asH_1(ID_i), P) \\ &= e(h_i H_1(ID_i), Q)e(W, pk_i)e(T, R_i)e(aH_1(ID_i), Q) \\ &= e((h_i + a)H_1(ID_i), Q)e(W, pk_i)e(T, R_i^*) \\ &= e(h_i^* H_1(ID_i), Q)e(W, pk_i)e(T, R_i^*) \end{aligned} \tag{4}$$

Observe that E_1 occurs in Game 1 or in Game 2. It follows that Jin-Wen's certificateless multi-proxy signature scheme is still insecure.

Furthermore, by the following steps, $\mathcal{A}$ is also able to forge a valid multi-proxy signature σ_{mp}^* on a message m^* under the warrant w^*.

1. Similar to the above process, without the knowledge of the original signer's secret value x_0, $\mathcal{A}$ is also able to obtain the pair (R_0^*, V_0^*) such that

$$e(V_0^*, P) = e(h_0^* H_1(ID_0), Q)e(W, pk_0)e(T, R_0^*) \tag{5}$$

   where $W = H_2(PP)$, $T = H_3(Q)$ and $h_0^* = H_5(PP||w^*||ID_0||pk_0||R_0)$. Then, $\mathcal{A}$ makes Public-Key-Inquiry, Partial-Private-Key-Extraction (this is omitted in Game 2) and Secret-Value-Extraction queries on each user $ID_i (i \in [n])$, respectively.
2. To sign the message m^* under the warrant w^* on behalf of ID_0, $\mathcal{A}$ with multi-proxy secret key $mpsk_i = (x_i, D_i, R_0^*, V_0^*) (i \in [n])$ performs the following steps.
   - Choose a random number $r_i \in \mathbb{Z}_q^*$ and compute $R_i^* = r_i P$.
   - Compute $V_i^* = h_i^* D_i + x_i W + r_i T$, where $W = H_2(PP)$, $T = H_3(Q)$ and $h_i^* = H_6(PP||m^*||ID_i||pk_i||R_i^*)$. It is clear that

$$e(V_i^*, P) = e(h_i^* H_1(ID_i), Q)e(W, pk_i)e(T, R_i^*) \tag{6}$$

– Compute $V_{mp}^* = \sum_{i=0}^{n} V_i^*$ and set $R_{mp}^* = (R_0^*, R_1^*, \ldots, R_n^*)$.

Combining the equation (5) and the equation (6), we have that

$$e(V_{mp}^*, P) = e(\sum_{i=0}^{n} h_i^* H_1(ID_i), Q) e(W, \sum_{i=0}^{n} pk_i) e(T, \sum_{i=0}^{n} R_i^*) \tag{7}$$

where $h_0^* = H_5(PP||w^*||ID_0||pk_0||R_0^*)$ and $h_i^* = H_6(PP||m^*||ID_i||pk_i||R_i^*)(i \in [n])$. Therefore, $\sigma_{mp}^* = (w^*, R_{mp}^*, V_{mp}^*)$ is a valid multi-proxy signature on the message m^* under the warrant w^*. Notice that the events E_2 and E_3 in Game 1 or in Game 2 both occur. That is the polynomial time adversary $\mathcal{A}$ wins the Game 1 (Game 2) with unit probability.

The reason of these attacks are successful is that the basic signature scheme proposed by Jin and Wen [9] is not secure and the multi-proxy signature is just a simple aggregation of the ordinary signatures generated by the participants. Thus, an adversary can forge a multi-proxy signature using the above approach. Next, we present an improved scheme to remedy the weaknesses in Jin-Wen's scheme.

4.3 Our improved scheme

Setup, **Partial-Private-Key-Extract** and **User-Key-Generate** are the same as those in Section 3 except for $H_4, H_5 : \{0,1\}^* \to \mathbb{G}$.

Sign: On input a message $m \in \{0,1\}^*$, the signer ID_i with private key sk_i does the following steps:

1. Choose $r \in \mathbb{Z}_q^*$ uniformly at random and compute $R = rP$.
2. Compute $W = H_2(PP)$ and $U = H_3(PP||m||ID_i||pk_i||R)$.
3. Compute $V = D_i + x_i W + rU$.

The signature on m is $\sigma = (R, V)$.

Verify: To verify a signature $\sigma = (R, V)$ on message m with respect to identity ID_i and the public key pk_i, the verifier checks whether $e(V, P) = e(H_1(ID_i), Q)e(W, pk_i)e(U, R)$, where $U = H_3(PP||m||ID_i||pk_i||R)$ and $W = H_2(PP)$. If this is not the case, output 0; otherwise, output 1.

Proxy-Key-Generate:

1. **Delegation-generation:** To delegate the signing capability, the original signer ID_0 signs on the warrant w which specifies the proxy policy and the identities of the original signer and the proxy signers.
   – Choose a random number $r_0 \in \mathbb{Z}_q^*$ and compute $R_0 = r_0 P$.
   – Compute $W = H_2(PP)$, $U_0 = H_4(01||PP||w||ID_0||pk_0||R_0)$ and $V_0 = D_0 + x_0 W + r_0 U_0$.
   – Send (w, R_0, V_0) to each proxy signer $ID_i (i \in [n])$.
2. **Delegation-verification:** The proxy signer $ID_i (i \in [n])$ accepts the delegation if $e(V_0, P) = e(H_1(ID_0), Q)e(W, pk_0)e(U_0, R_0)$; otherwise, he terminates the protocol.

3. **Proxy-secret-key-generation:** If the proxy signer $ID_i (i \in [n])$ accepts the delegation, then he sets $mpsk_i = (sk_i, R_0, V_0)$ as his multi-proxy secret key.

MP-Sign: On input a message $m \in \{0,1\}^*$ satisfying the warrant w and the identity ID_0 of the original signer, the proxy signer $ID_i (i \in [n])$ with multi-proxy secret key $mpsk_i$ does:

1. Choose a random number $r_i \in \mathbb{Z}_q^*$ and compute $R_i = r_i P$.
2. Compute $V_i = D_i + x_i W + r_i U_i$, where $U_i = H_5(11||PP||m||ID_i||pk_i||R_i)$ and $W = H_2(PP)$.
3. Send his partial proxy signature (w, R_0, V_0, R_i, V_i) to the clerk of the proxy group.

After receiving (w, R_0, V_0, R_i, V_i) from ID_i, the clerk checks the following equations:

$$e(V_0, P) = e(H_1(ID_0), Q)e(W, pk_0)e(U_0, R_0) \tag{8}$$

and

$$e(V_i, P) = e(H_1(ID_i), Q)e(W, pk_i)e(U_i, R_i) \tag{9}$$

where $U_0 = H_4(01||PP||w||ID_0||pk_0||R_0)$ and $U_i = H_5(11||PP||m||ID_i||pk_i||R_i)$ for any $i \in [n]$.

If all the equations hold, the clerk sets $\sigma_{mp} = (w, R_{mp}, V_{mp})$ as the multi-proxy signature on the message m under the warrant w, where $V_{mp} = \sum_{i=0}^{n} V_i$ and $R_{mp} = (R_0, R_1, \ldots, R_n)$.

MP-Verify: To verify the multi-proxy signature $\sigma_{mp} = (w, R_{mp}, V_{mp})$ on the message m under the warrant w, the verifier checks whether

$$e(V_{mp}, P) = e(\sum_{i=0}^{n} H_1(ID_i), Q)e(W, \sum_{i=0}^{n} pk_i) \prod_{i=0}^{n} e(U_i, R_i) \tag{10}$$

holds or not, where $W = H_2(PP)$, $U_0 = H_4(01||PP||w||ID_0||pk_0||R_0)$ and $U_i = H_5(11||PP||m||ID_i||pk_i||R_i)(i \in [n])$. If it holds, output 1; otherwise, output 0.

5 Conclusions

In this paper, by giving concrete attacks, we have indicated that Jin-Wen's certificateless multi-proxy signature scheme [9] is not secure according to their security model. We have also presented some corresponding improvements to prevent these attacks.

Acknowledgements

The authors thank Zhengping Jin and Fagen Li for their helpful comments. This work is supported by the Major Research Plan of the National Natural Science Foundation of China No. 90818005, the National Natural Science Foundation of China No. 60903217, No. 60773032 and by the Postdoctoral Science Foundation of China No. 20090450701.

References

1. Shamir A. Identity-based cryptosystems and signature schemes. CRYPTO84, LNCS 196. 1985. p.47–53.
2. Al-Riyami S, Paterson K. Certificateless public key cryptography. ASIACRYPT 2003, LNCS 2894. 2003. p.452–473.
3. Huang X, Susilo W, Mu Y, Zhang F. On the Security of Certificateless Signature Schemes from Asiacrypt 2003. CANS 2005, LNCS 3810. 2005. p.13–25.
4. Au M, Chen J, Liu J, Mu Y, Wong D, Yang G. Malicious KGC attacks in certificateless cryptography. ASIACCS 2007. 2007. p.302–311.
5. Zhang L, Zhang F. A new certificateless aggregate signature scheme. Computer Communications 2009;32(6):1079–1085.
6. Xiong H, Li F, Qin Z. A Provably Secure Proxy Signature Scheme in Certificateless Cryptography. Informatica 2010;21(2):277–294.
7. Liu Z, Hu Y, Zhang X, Ma H. Certificateless signcryption scheme in the standard model. Information Sciences 2010;180(3):452–464.
8. Weng J, Yao G, Deng H, Chen M, Li X. Cryptanalysis of a certificateless signcryption scheme in the standard model. Information Sciences 2011;181(3):661–667.
9. Jin Z, Wen Q. Certificateless multi-proxy signature. Computer Communications 2011;34(3):344-352.
10. Mambo M, Usuda K, Okamoto E. Proxy signature for delegating signing operation. Proceedings of the 3rd ACM Conference on Computer and Communications Security (CCS96). 1996. p.48–57.
11. Foster I, Kesselman C, Tsudik G, Tuecke S. A security architecture for computational grids. Proceedings of the 5th ACM Conference on Computers and Communications Security (CCS98). 1998. p.83–92.
12. Lee B, Kim H, Kim K. Strong proxy signature and its applications. Proceedings of Symposium on Cryptography and Information Security (SCIS 2001). 2001. p.603–608.
13. Park H, Lee I. A digital nominative proxy signature scheme for mobile communication. Proceedings of the 3rd International Conference on Information and Communications Security (ICICS 2001), LNCS 2229. 2001. p.451–4555.
14. Weissman J,Ramakrishnan S. Using Proxies to Accelerate Cloud Applications. Proceedings of the Workshop on Hot Topics in Cloud Computing. 2009. p.14–19.
15. Hwang S, Shi C. A simple multi-proxy signature scheme. Proceedings of the 10th National Conference on Information Security. 2000. p.134–138.
16. Cao F, Cao Z. A secure identity-based multi-proxy signature scheme. Computers and Electrical Engineering 2009;35(1):86–95.
17. Xiong H, Hua J, Chen Z, Li F. On the security of an identity based multi-proxy signature scheme. Computers and Electrical Engineering 2011;37(2):129–135.