Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2012, Article ID 416137, 13 pages
doi:10.1155/2012/416137

Research Article

On the Security of Some Aggregate Signature Schemes

Baoyuan Kang

School of Computer Science and Software, Tianjin Polytechnic University, Tianjin 300387, China

Correspondence should be addressed to Baoyuan Kang, baoyuankang@yahoo.com.cn

Received 11 September 2012; Revised 22 November 2012; Accepted 6 December 2012

Academic Editor: Tak-Wah Lam

Aggregate signature scheme proposed by Boneh, Gentry, Lynn, and Shacham allows n signatures on n distinct messages from n distinct users to aggregate a single signature that convince any verifier that n users did indeed sign the n messages, respectively. The main benefit of such schemes is that they allow bandwidth and computational savings. In this paper, we question about whether the existing aggregate signature schemes satisfy the basic property that they can convince any verifier that every user indeed signed the message which should be signed by him. We show that Rückert et al.'s scheme, and Shim's scheme do not satisfy the property. As a comparison, we investigate Boneh et al.'s scheme and show that under the assumption that each signer correctly signs one message, Boneh et al.'s scheme satisfies this property under two users' setting. Furthermore, we propose the concept of inside attack on aggregate signatures and give an improved aggregate signature scheme based on Shim's scheme. We also prove that the improved scheme is secure against inside attack.

1. Introduction

An aggregate signature scheme as introduced by Boneh et al. [1] is a method for combining n signatures from n different signers on n different messages into a single signature. This single signature (and the n original messages) will convince the verifier that the n signers did indeed sign the n original messages (i.e., signer i signed message m_i for $i = 1, \ldots, n$). Typical applications for aggregate signatures are, for example, secure routing [2] or certificate chain compression [1]. The main benefit of aggregate signature is that it saves bandwidth, which makes it an optimal solution for networks of small, battery-powered devices that communicate over energy-consuming wireless channels [3].

Since Boneh et al.'s aggregate signature scheme, many aggregate signature schemes are proposed [4–10]. There even are aggregate proxy signature [11] and aggregate signcryption schemes [12]. However, about the security of aggregate signature schemes, only traditional unforgeability was discussed in all existing schemes. We question that whether every existing aggregate signature satisfies the basic property proposed by Boneh et al. that it convinces any verifier that, for all $1 \leq i \leq n$, signer i indeed signed message m_i which should be signed by him; he didnot signed message m_j. Because in some situation an aggregate signature may satisfy the verification, even though signer i signed message m_j. We call this attack an inside attack on aggregate signatures. We think this is an important issue to aggregate signatures. Shao [13] discussed the security of aggregate signatures, but its issue was another aspect. He pointed that every signer i forges a signature $\sigma_i^{\cdot} = \sigma_i \cdot d_i$ on message m_i; here σ_i is the true signature of message m_i, when $d_1 \cdot d_2 \cdot \ldots \cdot d_n = 1$ and $S = \sigma_1^{\cdot} \cdot \sigma_2^{\cdot} \cdot \ldots \cdot \sigma_n^{\cdot}$ also satisfies the aggregate signature verification.

Recently, Rückert et al. [6] proposed the first aggregate signature in standard model. The scheme was based on the Boneh-Silverberg signature [14]. They proved its traditional unforgeability in the standard model while maintaining an optimal signature size and reasonable efficiency. However, in this paper, we show that Rückert et al.'s scheme does not satisfy the basic property that a verifier, given the aggregate signature along with the identities if the parties involved and their respective messages, can be convinced that signer i indeed signed message m_i which should be signed by him. In 2010, Shim proposed an efficient ID-based aggregate signature scheme with constant pairing computations [8]. It is the first scheme whose number of pairing computation in verification is independent of the number of users. But, in this paper we point that Shim's scheme also does not satisfy the basic property. As a comparison, we investigate Boneh et al.'s scheme [1] and show that under the assumption that each signer signs one message correctly, Boneh et al.'s scheme satisfies this property under two users' setting. Furthermore, we propose an improved scheme based on Shim's scheme and prove that the improved scheme is secure against the inside attack.

The rest of the paper is organized as follows. In Section 2 we introduce preliminaries and the computational assumption. Section 3 investigates the security of Rückert et al.'s aggregate signature. Section 4 investigates the security of the aggregate signature of Shim. As a comparison, we study Boneh et al.'s aggregate signature scheme in Section 5. The improved scheme is in Section 6. Section 7 concludes this paper.

2. Preliminary

2.1. The Bilinear Pairing

Let G_1 be a cyclic additive group generated by P, whose order is a prime q, and G_2 a cyclic multiplicative group of the same order. Let $e : G_1 \times G_1 \rightarrow G_2$ be a pairing map which satisfies the following conditions.

(1) Bilinearity: for any $P, Q, R \in G_1$, we have $e(P + Q, R) = e(P, R)e(Q, R)$ and

$$e(P, Q + R) = e(P, Q)e(P, R). \tag{2.1}$$

In particular, for any $a, b \in Z_q$, $e(aP, bP) = e(P, P)^{ab} = e(P, abP) = e(abP, P)$.

(2) Nondegeneracy: there exists $P, Q \in G_1$, such that $e(P, Q) \neq 1$.

(3) Computability: there is an efficient algorithm to compute $e(P, Q)$ for all $P, Q \in G_1$.

The typical way of obtaining such pairings is by deriving them from the Weil-pairing or the Tate-pairing on an elliptic curve over a finite field.

2.2. Gap Diffie-Hellman (GDH) Groups

Let G be a cyclic additive group of prime order q, and let P be a generator of G.

(1) The decisional Diffie-Hellman (DDH) problem is to decide whether $c = ab$ in Z/qZ for given $P, aP, bP, cP \in G$. If so, (P, aP, bP, cP) is called a valid Diffie-Hellman tuple.

(2) The computational Diffie-Hellman (CDH) problem is to compute abP for given $P, aP, bP \in G$.

Definition 2.1. The advantage of an algorithm F in solving the computational Diffie-Hellman problem on group G is

$$\mathrm{AdvCDH}_{\mathrm{F}} = \Pr\left[\mathrm{F}(P, aP, bP) = abP : \forall a, b \in Z_q\right]. \tag{2.2}$$

The probability took over the choice of a, b and F's coin tosses. An algorithm F is said to be (t, ε)-breaks the computational Diffie-Hellman problem on group G if F runs in time at most t, and $\mathrm{AdvCDH}_{\mathrm{F}}$ is at least ε.

Definition 2.2. A group G is said to be (t, ε)-gap Diffie-Hellman (GDH) group if the decisional Diffie-Hellman problem in G can be efficiently computable and there exists no algorithm (t, ε)-breaks the computational Diffie-Hellman problem on group G.

2.3. Security Model of Aggregate Signature

We take identity-based aggregate signature (IBAS) for example to give the definition of aggregate signature and its security model. An identity-based aggregate signature is composed of five algorithms [5]: key generation by the private key generation center (PKG), private key extraction by the PKG for individual users, signing by an individual user, aggregation of multiple individual signatures, and verification of an identity-based aggregate signature.

KeyGen

Take a security parameter λ as input and output system parameters params and master key msk.

KeyExt

Take params, msk. and a user identity ID as input and output a user private key S_{ID}.

Sign

Take private key S_{ID} and a message M as input and output an individual identity-based signature σ_{ID}.

Agg

Given n signatures $(\sigma_1, \ldots, \sigma_n)$ along with n users' identities $(\mathrm{ID}_1, \ldots, \mathrm{ID}_n)$ and n messages $(M_1, \ldots, M_n)$, output an aggregate signature σ_{Agg}.

Verify

Given an aggregate signature σ_{Agg}, the message, and identities' pair list $\{(M_1, \mathrm{ID}_1), \ldots, (M_n, \mathrm{ID}_n)\}$, verify the aggregate signature that if it is valid.

2.3.1. Security Model against Traditional Existential Forgery Attack

An IBAS scheme should be secure against traditional existential forgery under an adaptive chosen-message and an adaptive-chosen-identity attack. We formalize the security model as follows. The adversary's goal is the existential forgery of an aggregate signature. We give the adversary the power to choose the identities on which it wishes to forge a signature, the power to request the identity-based private key on all but one of these identities. The adversary's advantage is defined as its probability of success in the following game.

Setup. The adversary is given the needed parameters and an identity ID_1 at random.

Extraction Queries

Given an identity ID_i $(i \neq 1)$, the challenger returns the private key S_{ID_i} corresponding to ID_i.

Signature Queries

Proceeding adaptively, the adversary may request signatures with respect to identity ID_i on messages of his choice.

Response

Finally, the adversary outputs $n - 1$ additional identities $(\mathrm{ID}_2, \ldots, \mathrm{ID}_n)$, n messages $(M_1, \ldots, M_n)$ and an aggregate signature σ with respect to these n identities, and n messages $(M_1, \ldots, M_n)$.

The adversary wins if the aggregate signature σ is a valid signature on $(M_1, \ldots, M_n)$ under $\mathrm{ID}_1, \mathrm{ID}_2, \ldots, \mathrm{ID}_n$ and the adversary did not request the private key for ID_1 and did not request a signature on M_1 under ID_1.

2.3.2. Security Model against Inside Existential Forgery Attack

We defined one new secure concept of aggregate signature as inside attack. It means the included signers to generate an aggregate signature V on messages $(M_{j_1}, M_{j_2} \ldots, M_{j_n})$ for identities $(\mathrm{ID}_1, \ldots, \mathrm{ID}_n)$. But, they claim that they generate an aggregate signature V on messages $(M_1, M_2 \ldots, M_n)$ for identities $(\mathrm{ID}_1, \ldots, \mathrm{ID}_n)$, here

(1) $(M_1, M_2 \ldots, M_n) \neq (M_{j_1}, M_{j_2} \ldots, M_{j_n})$,

(2) V really satisfies the aggregate signature verification equation on messages $(M_1, M_2 \ldots, M_n)$ for identities $(\mathrm{ID}_1, \ldots, \mathrm{ID}_n)$.

The concept of inside attack is closely related to the basic property of aggregate signature that it should convince any verifier that every user indeed signed the message which should be signed by him.

3. The Security of the Aggregate Signature Rückert et al.'s Scheme

3.1. Brief Review of Rückert et al.'s Scheme

In Rückert et al.'s scheme [6], two groups G_1 and G_2 of prime order l and a multilinear map e are used; g is a generator of G_1. If $a_1, \ldots, a_n \in Z$, and $x_1, \ldots, x_n \in G_1$, then $e(x_1^{a_1}, x_2^{a_2}, \ldots, x_n^{a_n}) = e(x_1, x_2, \ldots, x_n)^{a_1 a_2 \cdots a_n}$. Rückert et al.'s scheme comprises five algorithms.

Key Generation

The key generation algorithm takes as input the security parameter. It randomly selects $2n$ elements $a_{1,0}, a_{1,1}, \ldots, a_{n,0}, a_{n,1} \in \{1, \ldots, l-1\}$. The algorithm computes

$$u_{1,0} = g^{a_{1,0}}, \qquad u_{1,1} = g^{a_{1,1}}, \ldots, u_{n,0} = g^{a_{n,0}}, \qquad u_{n,1} = g^{a_{n,1}} \tag{3.1}$$

and returns the private key and the public key pair:

$$\mathrm{sk} = (a_{1,0}, a_{1,1}, \ldots, a_{n,0}, a_{n,1}), \qquad \mathrm{pk} = (u_{1,0}, u_{1,1}, \ldots, u_{n,0}, u_{n,1}). \tag{3.2}$$

Signature Issue

It accepts as input a message $m = (m_1, \ldots, m_n) \in \{0,1\}^n$ as well as signing key $\mathrm{sk} = (a_{1,0}, a_{1,1}, \ldots, a_{n,0}, a_{n,1})$ and computes the signature $\sigma = g^{\prod_{i=1}^{n} a_{i,m_i}}$.

Signature Verification

It returns 1 iff $e(\sigma, g, \ldots, g) = e(u_{1,m_1}, u_{2,m_2}, \ldots, u_{n,m_n})$.

Signature Aggregation

It builds an aggregate signature S on messages $m^{(1)}, \ldots, m^{(q)}$, under public keys $\mathrm{pk}^{(1)}, \ldots, \mathrm{pk}^{(q)}$, respectively. It outputs the triple (pk, M, S). Here $S = \prod_{i=1}^{q} \sigma^{(i)}$, $\mathrm{pk} = \{\mathrm{pk}^{(1)}, \ldots, \mathrm{pk}^{(q)}\}$, $M = \{m^{(1)}, \ldots, m^{(q)}\}$, and $\sigma^{(i)}$ is the signature on message $m^{(i)}$ produced by the user with public key $\mathrm{pk}^{(i)}$.

Aggregate Verification

It takes as input a set of public keys $\mathrm{pk} = \{\mathrm{pk}^{(1)}, \ldots, \mathrm{pk}^{(q)}\}$, a set of messages $M = \{m^{(1)}, \ldots, m^{(q)}\}$, and an aggregate signature S. It returns 1 iff

$$\prod_{i=1}^{q} e\left(u^{(i)}_{1,m_1^{(i)}}, u^{(i)}_{2,m_2^{(i)}}, \ldots, u^{(i)}_{n,m_n^{(i)}}\right) = e(S, g, \ldots, g). \tag{3.3}$$

3.2. The Security of Rückert et al.'s Scheme

In Rückert et al.'s scheme, let $n = 2$, two users A_1, A_2 with private key and pubic key pairs:

$$\begin{aligned} \mathrm{sk}^{(1)} &= \left(a^{(1)}_{1,0}, a^{(1)}_{1,1}, a^{(1)}_{2,0}, a^{(1)}_{2,1}\right), \qquad \mathrm{pk}^{(1)} = \left(u^{(1)}_{1,0}, u^{(1)}_{1,1}, u^{(1)}_{2,0}, u^{(1)}_{2,1}\right), \\ \mathrm{sk}^{(2)} &= \left(a^{(2)}_{1,0}, a^{(2)}_{1,1}, a^{(2)}_{2,0}, a^{(2)}_{2,1}\right), \qquad \mathrm{pk}^{(2)} = \left(u^{(2)}_{1,0}, u^{(2)}_{1,1}, u^{(2)}_{2,0}, u^{(2)}_{2,1}\right), \end{aligned} \tag{3.4}$$

respectively.

Let $m^{(1)} = (m_1^{(1)}, m_2^{(1)})$, $m^{(2)} = (m_1^{(2)}, m_2^{(2)})$ be two messages. Then $\sigma^{(1)} = g^{a^{(1)}_{1,m_1^{(1)}} \cdot a^{(1)}_{2,m_2^{(1)}}}$ is the signature on $m^{(1)}$ by A_1, $\sigma^{(2)} = g^{a^{(2)}_{1,m_1^{(2)}} \cdot a^{(2)}_{2,m_2^{(2)}}}$ is the signature on $m^{(2)}$ by A_2. So the aggregate signature produced by users A_1, A_2 is

$$S = \sigma^{(1)} \cdot \sigma^{(2)} = g^{a^{(1)}_{1,m_1^{(1)}} \cdot a^{(1)}_{2,m_2^{(1)}} + a^{(2)}_{1.m_1^{(2)}} \cdot a^{(2)}_{2,m_2^{(2)}}}. \tag{3.5}$$

The aggregate verification equation

$$e\left(u^{(1)}_{1,m_1^{(1)}}, u^{(1)}_{2,m_2^{(1)}}\right) \cdot e\left(u^{(2)}_{1,m_1^{(2)}}, u^{(2)}_{2,m_2^{(2)}}\right) = e(g, g)^{a^{(1)}_{1,m_1^{(1)}} \cdot a^{(1)}_{2,m_2^{(1)}} + a^{(2)}_{1,m_1^{(2)}} \cdot a^{(2)}_{2,m_2^{(2)}}} = e(S, g) \tag{3.6}$$

holds.

However, when $m_1^{(1)} = m_1^{(2)}$, $m_2^{(1)} = 0$, $m_2^{(2)} = 1$, $a^{(1)}_{1,m_1^{(1)}} = a^{(2)}_{1,m_1^{(2)}} = 1$, and $a^{(1)}_{2,0} + a^{(2)}_{2,1} = a^{(2)}_{2,0} + a^{(1)}_{2,1}$, The equation

$$a^{(1)}_{1,m_1^{(1)}} \cdot a^{(1)}_{2,m_2^{(1)}} + a^{(2)}_{1,m_1^{(2)}} \cdot a^{(2)}_{2,m_2^{(2)}} = a^{(1)}_{1,m_1^{(2)}} \cdot a^{(1)}_{2,m_2^{(2)}} + a^{(2)}_{1,m_1^{(1)}} \cdot a^{(2)}_{2,m_2^{(1)}}, \tag{3.7}$$

holds. So when the user with public key $\mathrm{pk}^{(1)}$ signs $m^{(2)}$, the user with public key $\mathrm{pk}^{(2)}$ signs $m^{(1)}$, they generate aggregate signature

$$S^* = g^{a^{(1)}_{1,m_1^{(2)}} \cdot a^{(1)}_{2,m_2^{(2)}} + a^{(2)}_{1.m_1^{(1)}} \cdot a^{(2)}_{2,m_2^{(1)}}}, \tag{3.8}$$

and also satisfies the aggregate verification equation

$$e\left(u^{(1)}_{1,m_1^{(1)}}, u^{(1)}_{2,m_2^{(1)}}\right) \cdot e\left(u^{(2)}_{1,m_1^{(2)}}, u^{(2)}_{2,m_2^{(2)}}\right) = e(S^*, g). \tag{3.9}$$

In this situation, the aggregate signature cannot convince the verifier that signer i signed message m_i. So Rückert et al.'s aggregate signature is not secure; it does not satisfy the property that a verifier, given the aggregate signature along with the identities if the parties involved and their respective messages, can be convinced that signer i indeed signed message m_i which should be signed by him. It is not secure against the inside forgery attack.

4. The Security of Shim's Aggregate Signature Scheme

4.1. Brief Review of Shim's Scheme

Shim's scheme [8] comprises five algorithms.

Setup. Given security parameter $k \in Z$, the algorithm works as follows.

(1) Generate a prime q, a cyclic additive group G_1 and a cyclic multiplicative group G_2 of prime order q, a generator P in G_1 and an admissible pairing $e : G_1 \times G_1 \to G_2$.

(2) Pick a random $s \in Z_q^{\cdot}$ and set $P_{\text{pub}} = sP$.

(3) Choose cryptographic hash functions $H_1 : \{0,1\}^* \to G_1$ and $H_2 : \{0,1\}^* \to Z_q$.

The system parameters are $\langle q, G_1, G_2, e, P, P_{\text{pub}}, H_1, H_2 \rangle$.

Extract

For a given string $\text{ID} \in \{0,1\}^*$.

(1) Compute $Q_{\text{ID}} = H_1(\text{ID}) \in G_1$.

(2) Set the private key S_{ID} to be $s \cdot Q_{\text{ID}}$, where s is a master secret.

Sign

Given a private S_{ID} and a message $M \in \{0,1\}^*$.

(1) Choose $r \in_R Z_q^*$ and compute $U = r \cdot P \in G_1$.

(2) Compute $h = H_2(\text{ID}, M, U) \in Z_q$ and $V = S_{\text{ID}} + h \cdot r \cdot P_{\text{pub}} \in G_1$. The signature on M is $\sigma = (U, V)$.

Agg

For the aggregating set of users S, assign to each user an index i, ranging from 1 to $k = |S|$.

(1) Each user $A_i \in S$ computes signature $\sigma_i = (U_i, V_I)$ on a message $M_i \in \{0,1\}^*$.

(2) Compute $V = \sum_{i=1}^{k} V_i$ and output $\sigma = (U_1, \ldots, U_k, V)$ as an aggregate signature on $(M_1, \ldots, M_k)$ for $(\text{ID}_1, \ldots, \text{ID}_k)$.

AVerify

Given an aggregate signature $\sigma = (U_1, \ldots, U_k, V)$ as above.

(1) Compute $Q_i = H_1(\mathrm{ID}_i)$ and $h_i = H_2(\mathrm{ID}_i, M_i, U_i)$ for $i = 1, \ldots, k$.

(2) Verify whether $e(V, P) = e(\sum_{i=1}^{k}[Q_i + h_i \cdot U_i], P_{\mathrm{pub}})$ holds or not. If it holds, accept the aggregate signature $\sigma = (U_1, \ldots, U_k, V)$.

4.2. Attack on Shim's Scheme

Let ID_1 be an identity of signer A_1 and let ID_2 be an identity of signer A_2. They claim that they generate an aggregate signature $\sigma = (U_1, U_2, V)$ on messages (M_1, M_2) for identities $(\mathrm{ID}_1, \mathrm{ID}_2)$. Then, A_1 should sign M_1, and A_2 should sign M_2. That is to say, they should do as following:

(1) A_1 and A_2 choose $r_1, r_2 \in_R Z_q^*$ and compute $U_1 = r_1 \cdot P$ and $U_2 = r_2 \cdot P$, respectively.

(2) A_1 and A_2 compute

$$V_1 = S_1 + H_2(\mathrm{ID}_1, M_1, U_1) \cdot r_1 \cdot P_{\mathrm{pub}}, \qquad V_2 = S_2 + H_2(\mathrm{ID}_2, M_2, U_2) \cdot r_2 \cdot P_{\mathrm{pub}}, \tag{4.1}$$

respectively.

(3) They generate aggregate signature $\sigma = (U_1, U_2, V)$ on messages (M_1, M_2) for identities $(\mathrm{ID}_1, \mathrm{ID}_2)$. Here $V = V_1 + V_2$.

But, if the aggregate signature satisfies the verification equation, can the verifier be convinced that A_1 indeed has signed M_1, and A_2 indeed has signed M_2? They may cooperate to do on purpose as following:

(1) A_1 and A_2 Choose $r_1, r_2 \in_R Z_q^*$ and compute $U_1 = r_1 \cdot P$ and $U_2 = r_2 \cdot P$.

(2) A_1 and A_2 compute

$$V_1^* = S_1 + H_2(\mathrm{ID}_2, M_2, U_2) \cdot r_2 \cdot P_{\mathrm{pub}}, \qquad V_2^* = S_2 + H_2(\mathrm{ID}_1, M_1, U_1) \cdot r_1 \cdot P_{\mathrm{pub}}, \tag{4.2}$$

respectively. They have not signed M_1 and M_2, respectively.

(3) They claim that they generate aggregate signature $\sigma = (U_1, U_2, V^*)$ on messages (M_1, M_2) for identities $(\mathrm{ID}_1, \mathrm{ID}_2)$. Here $V^* = V_1^* + V_2^*$.

Since $V^* = V_1^* + V_2^* = V_1 + V_2 = V$, the verification equation

$$e(V^*, P) = e\big((Q_1 + h_1 U_1 + Q_2 + h_2 U_2), P_{\mathrm{pub}}\big). \tag{4.3}$$

Holds. A_1 and A_2 succeed in forging aggregate signature for $(\mathrm{ID}_1, \mathrm{ID}_2)$ on (M_1, M_2).

The weakness of Shim's scheme against this inside forgery attack is due to the separation of the message signed and the private key in the signing equation $V = S_{\mathrm{ID}} + h \cdot r \cdot P_{\mathrm{pub}} \in G_1$.

5. The Security of Boneh et al.'s Aggregate Schemes

We can investigate the security of Boneh et al.'s aggregate signature scheme [1] to provide further illustration to this flaw of about two schemes.

5.1. Brief Review of Boneh et al.'s Scheme

In Boneh et al.'s aggregate signature, two cyclic multiplicative groups G_1 and G_2 of prime order and a bilinear map $e : G_1 \times G_1 \rightarrow G_2$ are used. g is a generator of G_1. The scheme employs a hash function $h : \{0,1\}^{\cdot} \rightarrow G_2$.

Boneh et al.'s aggregate signature scheme comprises five algorithms.

Key Generation

For a user, pick random $x \leftarrow Z_p$, and compute $v = g^x$. The user's public key is $v \in G_1$, and secret key is $x \in Z_p$.

Signing

Given the secret key x and a message $m \in \{0,1\}^{\cdot}$, compute $h = h(m)$, and the signature $\sigma = h^x$.

Verification

Given user's public key v, a message m, and a signature σ, compute $h = h(m)$; accept if $e(g,\sigma) = e(v,h)$ holds.

Aggregation

For the aggregating set of users U, assign to each user an index i, ranging from 1 to $k = |U|$. Each user $u_i \in U$ provides a signature $\sigma_i \in G_2$ on a message $m_i \in \{0,1\}^{\cdot}$ of his choice. Compute the aggregate signature $\sigma = \prod_{i=1}^{k} \sigma_i$.

Aggregate Verification

Given an aggregate signature σ for an aggregating set of users U, indexed as before, and given the original messages $m_i \in \{0,1\}^{\cdot}$ and public keys v_i for all users $u_i \in U$. Compute $h_i = h(m_i)$ for $1 \le i \le k$, and accept if $e(g,\sigma) = \prod_{i=1}^{k} e(v_i, h_i)$ holds.

5.2. The Security of Boneh et al.'s Scheme

In Boneh et al.'s scheme, given an aggregate signature of two different messages m_1 and m_2 under two users with public keys v_1 and v_2, respectively, if

$$e(v_1, h_1) \cdot e(v_2, h_2) = e(v_2, h_1) \cdot e(v_1, h_2), \tag{5.1}$$

then, it will be impossible to know whether signer i signed message m_i, and Boneh et al.'s scheme will have the same flaw as that of Rückert et al.'s scheme. But if

$$e(v_1, h_1) \cdot e(v_2, h_2) = e(v_2, h_1) \cdot e(v_1, h_2), \tag{5.2}$$

then

$$\begin{aligned} e(g^{x_1}, h_1) \cdot e(g^{x_2}, h_2) &= e(g^{x_2}, h_1) \cdot e(g^{x_1}, h_2), \\ e(g, h_1^{x_1}) \cdot e(g, h_2^{x_2}) &= e(g, h_1^{x_2}) \cdot e(g, h_2^{x_1}), \\ e(g, h_1^{x_1} h_2^{x_2}) &= e(g, h_1^{x_2} h_2^{x_1}), \\ h_1^{x_1} h_2^{x_2} &= h_1^{x_2} h_2^{x_1}, \\ h_1^{x_1 - x_2} &= h_2^{x_1 - x_2}, \\ h_1 &= h_2, \\ h(m_1) &= h(m_2). \end{aligned} \tag{5.3}$$

So if the hash function h is secured, $h(m_1) \neq h(m_2)$, then, under the assumption that each signer signs one message correctly, Boneh et al.'s scheme does not suffer the same flaw as about two schemes under two users.

6. An Improvement of Shim's Identity-Based Aggregate Signature Scheme

6.1. The Improved Scheme

The improved scheme comprises five algorithms.

Setup. Given security parameter $k \in Z$, the algorithm works as follows.

(1) Generate a prime q, a cyclic additive group G_1 and a cyclic multiplicative group G_2 of prime order q, two random generators P and Q in G_1, and an admissible pairing $e : G_1 \times G_1 \to G_2$.

(2) Pick a random $s \in Z_q^{\cdot}$ and set $P_{\text{pub}} = sP$.

(3) Choose cryptographic hash functions $H_1 : \{0,1\}^* \to G_1$ and $H_2 : \{0,1\}^* \to Z_q$.

The system parameters are $\langle q, G_1, G_2, e, P, P_{\text{pub}}, H_1, H_2 \rangle$.

Extract

For a given string $\text{ID} \in \{0,1\}^*$.

(1) Compute $Q_{\text{ID}} = H_1(\text{ID}) \in G_1$.

(2) Set the private key S_{ID} to be $s \cdot Q_{\text{ID}}$, where s is a master secret.

Sign

Given a private S_{ID} and a message $M \in \{0,1\}^*$.

(1) Choose $r \in_R Z_q^*$ and compute $U = r \cdot P \in G_1$.

(2) Compute $h = H_2(\mathrm{ID}, M, U) \in Z_q$ and $V = hS_{\mathrm{ID}} + r \cdot Q \in G_1$. The signature on M is $\sigma = (U, V)$.

Agg

For the aggregating set of users S, assign to each user an index i, ranging from 1 to $k = |S|$.

(1) Each user $A_i \in S$ computes signature $\sigma_i = (U_i, V_I)$ on a message $M_i \in \{0,1\}^*$.

(2) Compute $V = \sum_{i=1}^{k} V_i$ and output $\sigma = (U_1, \ldots, U_k, V)$ as an aggregate signature on $(M_1, \ldots, M_k)$ for $(\mathrm{ID}_1, \ldots, \mathrm{ID}_k)$.

AVerify

Given an aggregate signature $\sigma = (U_1, \ldots, U_k, V)$ as above.

(1) Compute $Q_i = H_1(\mathrm{ID}_i)$ and $h_i = H_2(\mathrm{ID}_i, M_i, U_i)$ for $i = 1, \ldots, k$.

(2) Verify whether $e(V, P) = e(P_{\mathrm{pub}}, \sum_{i=1}^{n} h_i Q_{\mathrm{ID}_i}) e(Q, \sum_{i=1}^{n} U_i)$ holds or not. If it holds, accept the aggregate signature $\sigma = (U_1, \ldots, U_k, V)$.

6.2. Security of the Improved Scheme

Following the method in [10], it is easy to prove that the improved scheme is secure against the traditional existential forgery under an adaptive chosen message and an adaptive-chosen identity attack. Here, we only show that our improvement is secure against the inside attack proposed by us.

Take two signers as example, let ID_1 be the identity of signer A_1, and ID_2 the identity of signer A_2. If they cooperate to do as following:

(1) A_1 and A_2 Choose $r_1, r_2 \in_R Z_q^*$ and compute $U_1 = r_1 \cdot P$ and $U_2 = r_2 \cdot P$.

(2) A_1 and A_2 compute

$$V_1^* = H_2(\mathrm{ID}_2, M_2, U_2) S_{\mathrm{ID}_1} + r_2 Q, \qquad V_2^* = H_2(\mathrm{ID}_1, M_1, U_1) S_{\mathrm{ID}_2} + r_1 Q, \tag{6.1}$$

respectively. Note that they have not signed M_1 and M_2, respectively.

(3) They claim that they generate aggregate signature $\sigma = (U_1, U_2, V^*)$ on messages (M_1, M_2) for identities $(\mathrm{ID}_1, \mathrm{ID}_2)$. Here $V^* = V_1^* + V_2^*$.

But, when $\sigma = (U_1, U_2, V^*)$ is a valid aggregate signature on messages (M_1, M_2) for identities $(\mathrm{ID}_1, \mathrm{ID}_2)$, the following equation holds:

$$e(V^*, P) = e\big(H_2(\mathrm{ID}_1, M_1, U_1) Q_{\mathrm{ID}_1} + H_2(\mathrm{ID}_2, M_2, U_2) Q_{\mathrm{ID}_2}, P_{\mathrm{pub}}\big) e(Q, U_1 + U_2). \tag{6.2}$$

In fact

$$\begin{aligned} e(V^{*}, P) &= e(H_2(\mathrm{ID}_2, M_2, U_2)S_{\mathrm{ID}_1} + r_2Q + H_2(\mathrm{ID}_1, M_1, U_1)S_{\mathrm{ID}_2} + r_1Q, P) \\ &= e\big(H_2(\mathrm{ID}_2, M_2, U_2)Q_{\mathrm{ID}_1} + H_2(\mathrm{ID}_1, M_1, U_1)Q_{\mathrm{ID}_2}, P_{\mathrm{pub}}\big)e(Q, U_1 + U_2). \end{aligned} \tag{6.3}$$

If

$$\begin{aligned} &e\big(H_2(\mathrm{ID}_1, M_1, U_1)Q_{\mathrm{ID}_1} + H_2(\mathrm{ID}_2, M_2, U_2)Q_{\mathrm{ID}_2}, P_{\mathrm{pub}}\big)e(Q, U_1 + U_2) \\ &\quad = e\big(H_2(\mathrm{ID}_2, M_2, U_2)Q_{\mathrm{ID}_1} + H_2(\mathrm{ID}_1, M_1, U_1)Q_{\mathrm{ID}_2}, P_{\mathrm{pub}}\big)e(Q, U_1 + U_2), \end{aligned} \tag{6.4}$$

then

$$\begin{aligned} &e\big(H_2(\mathrm{ID}_1, M_1, U_1)Q_{\mathrm{ID}_1} + H_2(\mathrm{ID}_2, M_2, U_2)Q_{\mathrm{ID}_2}, P_{\mathrm{pub}}\big) \\ &\quad = e\big(H_2(\mathrm{ID}_2, M_2, U_2)Q_{\mathrm{ID}_1} + H_2(\mathrm{ID}_1, M_1, U_1)Q_{\mathrm{ID}_2}, P_{\mathrm{pub}}\big). \end{aligned} \tag{6.5}$$

So

$$\begin{gathered} H_2(\mathrm{ID}_1, M_1, U_1)Q_{\mathrm{ID}_1} + H_2(\mathrm{ID}_2, M_2, U_2)Q_{\mathrm{ID}_2} = H_2(\mathrm{ID}_2, M_2, U_2)Q_{\mathrm{ID}_1} \\ + H_2(\mathrm{ID}_1, M_1, U_1)Q_{\mathrm{ID}_2}, \\ (H_2(\mathrm{ID}_2, M_2, U_2) - H(\mathrm{ID}_1, M_1, U_1))Q_{\mathrm{ID}_2} = (H_2(\mathrm{ID}_2, M_2, U_2) - H(\mathrm{ID}_1, M_1, U_1))Q_{\mathrm{ID}_1} \\ Q_{\mathrm{ID}_2} = Q_{\mathrm{ID}_1}. \end{gathered} \tag{6.6}$$

This is impossible. So the inside attack is not successful in improved scheme in two signers' setting.

In n signers' setting, if they generate an aggregate signature V on messages $(M_{j_1}, M_{j_2} \ldots, M_{j_n})$ for identities $(\mathrm{ID}_1, \ldots, \mathrm{ID}_n)$. But they claim that they generate an aggregate signature V on messages $(M_1, M_2 \ldots, M_n)$ for identities $(\mathrm{ID}_1, \ldots, \mathrm{ID}_n)$, here $(M_1, M_2 \ldots, M_n) \neq (M_{j_1}, M_{j_2} \ldots, M_{j_n})$. Then the probability of V satisfying the aggregate signature verification equation on messages $(M_1, M_2 \ldots, M_n)$ for identities $(\mathrm{ID}_1, \ldots, \mathrm{ID}_n)$ is equal to the probability of the following equation holding

$$\begin{aligned} &\big(H_2\big(\mathrm{ID}_{j_1}, M_{j_1}, U_{j_1}\big) - H(\mathrm{ID}_1, M_1, U_1)\big)Q_{\mathrm{ID}_1} + \big(H_2\big(\mathrm{ID}_{j_2}, M_{j_2}, U_{j_2}\big) - H(\mathrm{ID}_2, M_2, U_2)\big)Q_{\mathrm{ID}_2} \\ &\quad + \cdots + \big(H_2\big(\mathrm{ID}_{j_n}, M_{j_n}, U_{j_n}\big) - H(\mathrm{ID}_n, M_n, U_n)\big)Q_{\mathrm{ID}_n} = \mathrm{O}. \end{aligned} \tag{6.7}$$

Here O denotes the identity of the cyclic additive group G_1. So the improved aggregate signature scheme is secured against the inside attack.

7. Conclusion

In this paper, we analyse the security of some aggregate signature schemes. We show that Rückert et al.'s scheme cannot convince the verifier that every signer indeed signed the message which should be signed by him. Shim's scheme also suffers such flaw. As a comparison, we investigate Boneh et al.'s scheme and show that under the assumption that each signer signs one message correctly, Boneh et al.'s aggregate scheme can convince the verifier that every signer indeed signed the message which should be signed by him under two users. Furthermore, we propose the concept of inside attack on aggregate signatures and give an improved scheme based on Shim's scheme. We also prove that the improved scheme is secured against the inside attack.

References

[1] D. Boneh, C. Gentry, H. Shacham, and B. Lynn, "Aggregate and verifiably encrypted signatures from bilinear maps," in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRPYT '03)*, vol. 2656 of *Lecture Notes in Computer Science*, pp. 416–432, Springer, Warsaw, Poland, May 2003.

[2] S. T. Kent, C. Lynn, J. Mikkelson, and K. Seo, "Secure border gateway protocol (S-BGP)-real world performance and deployment issues," in *Proceedings of the Network and Distributed System Security Symposium (NDSS '00)*, Internet Society, 2000.

[3] G. Neven, "Efficient sequential aggregate signed date," in *Proceedings of the 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRPYT '08)*, vol. 4965 of *Lecture Notes in Computer Science*, pp. 52–69, Springer, 2008.

[4] X. Cheng, J. Liu, and X. Wang, "Identity-based aggregate and verifiably encrypted signatures from bilinear pairing," in *Proceedings of the International Conference on Computational Science and Its Applications (ICCSA '05)*, vol. 3483, pp. 1046–1054, May 2005.

[5] C. Gentry and Z. Ramzan, "Identity-based aggregate signature," in *Proceedings the 9th International Workshop on Theory and Practice in Public Key Cryptography (PKC '06)*, vol. 3958 of *Lecture Notes in Computer Science*, pp. 257–273, Springer, 2006.

[6] M. Rückert and D. Schröde, "Aggregate and verifiably encrypted signatures from multilinear maps without random oracles," in *Proceedings of the the 3rd International Conference on Information Security and Assurance (ISA'09)*, vol. 5576 of *Lecture Notes in Computer Science*, pp. 750–759, Springer, 2009.

[7] R. Sakai, K. Ohgishi, and M. Kasahara, "Cryptosystems based on pairing," in *Proceedings of the Symposium on Cryptography and Information Security*, pp. 26–28, Okinawa, Japan, 2000.

[8] K. A. Shim, "An ID-based aggregate signature scheme with constant pairing computations," *The Journal of Systems and Software*, vol. 83, no. 10, pp. 1873–1880, 2010.

[9] L. Zhang and F. Zhang, "A new certificateless aggregate signature scheme," *Computer Communications*, vol. 32, no. 6, pp. 1079–1085, 2009.

[10] Y. Yu, X. Zheng, and H. Sun, "An identity based aggregate signature from pairings," *Journal of Networks*, vol. 6, no. 4, pp. 631–637, 2011.

[11] J. Li, K. Kim, F. Zhang, and X. Chen, "Aggregate proxy signature and verifiably encrypted proxy signature," in *Proceedings of the International Conference (ProvSec '07)*, vol. 4784 of *Lecture Notes in Computer Science*, pp. 208–217, Springer, 2007.

[12] S. Selvi, S. Vivek, J. Shriram, and S. Kalaivani, "Identity based aggregate signcryption schemes," in *Proceedings of the 10th International Conference on Cryptology in India (INDOCRYPT '09)*, vol. 5922 of *Lecture Notes in Computer Science*, pp. 378–397, Springer, 2009.

[13] Z. Shao :, "Enhanced aggregate signature from pairings," in *Proceedings of the Conference on Iformation Security and Cryptology (CISC '05)*, vol. 3822 of *Lecture Notes in Computer Science*, pp. 140–149, Springer, 2005.

[14] D. Boneh and A. Silverberg, "Applications of multilinear forms to cryptography," *Toptic in Algebraic and Noncommutative Geometry, Contemporary Mathematics*, vol. 324, pp. 71–90, 2003.