Two Identity-Based Aggregate Signature Schemes from Pairings

Subhas Chandra Sahana, Sourav Kumar Das, Sangeeta Mashahary and Bubu Bhuyan

Abstract An aggregate signature is a short digital signature which is the output of aggregation process. The signature aggregation is done on k signatures of k distinct messages from k distinct users. As the produced signature size is shorter, so it will be efficient to use the schemes in low-bandwidth communication environment. In this paper, we proposed two identity-based aggregate signature schemes from bilinear pairing operations. The proposed schemes are secure against existential forgery under adaptively chosen message and identity attack in the random oracle model based on the assumption of intractability of the computational Diffie–Hellman problem (CDHP). The efficiency analysis of the proposed identity-based aggregate signature schemes with other established identity-based aggregate signature schemes is also done in this paper.

Keywords Signature · Aggregate signature · Bilinear pairing · Computational Diffie–Hellman Problem (CDHP)

S. C. Sahana (✉) · S. K. Das · S. Mashahary · B. Bhuyan
Department of Information Technology, North Eastern
Hill University, Shillong 793022, India
e-mail: subhas.sahana@gmail.com

S. K. Das
e-mail: kumarsourav.it@gmail.com

S. Mashahary
e-mail: sangeetamashahary133@gmail.com

B. Bhuyan
e-mail: b.bhuyan@gmail.com

R. Bera et al. (eds.), *Advances in Communication, Devices and Networking*,
Lecture Notes in Electrical Engineering 462,
https://doi.org/10.1007/978-981-10-7901-6_94

1 Introduction

In 2003, the first aggregate signature (BGLS), proposed by Boneh et al. [1] allows k members of a given group of potential signers to sign k different messages and all these signatures can be aggregated into a single signature. Actually, the aggregate signature [1] is based on the BLS [2] short signature.

As the size of aggregate signature is same as the individual signature, so we get a compact single signature of all individual signatures. This single signature can provide a proof to the verifier that the n players have indeed signed the original messages. Thus, aggregate signature provides non-repudiation security service on different messages signed by different users at the same time. Actually, there have been many practical application of aggregate signature scheme. As we are bounded by page limitation, only one example has been discussed. In public key infrastructure (PKI) of depth n, each user has been given a chain of certificate of length n. So, the chain contains n signatures by n certificate authorities (CAs) on n distinct certificate. If we use aggregate signature scheme, it is possible to obtain a compressed aggregated certificate [3]. Specifically, the main motivation is that X.509 certificates can be shortened into a single signature by compressing n signatures. It is also useful for compression where the signatures on many different messages are generated by many different users [4].

It is well known that that PKI-based cryptosystem has the biggest disadvantage related to certificate management activities. To avoid this problem, Shamir [5] introduced the concept of identity-based cryptosystem (IBC) in 1984. In IBC, the main advantage is that there is no need of public key distribution in the form of certificates as user can use his unique identity information such as name, email address by providing his own public key.

Due to various interesting practical applications and various advantages of IBC, discussed above, it is always a hot research area to achieve an efficient Id-based aggregate signature schemes. After the pioneering work [1, 2], many identity-based aggregate signature schemes have been proposed. In 2004, Cheon [6] presented first identity-based aggregate signature (IBAS). This scheme compresses the signatures into half, while the BGLS compresses multiple signatures into one. After that work, in 2006, Gentry and Ramzan proposed an efficient ID-based aggregate signature which is much faster than BGLS scheme as less number of operations are involved. In 2008, Wang [7] presented a new ID-based aggregate scheme which provides partial aggregation. It is also more efficient than BGLS scheme. At the same time, Wen [8] proposed a new aggregate signature with constant pairing operation (AS-CPO) scheme, which requires only two pairings in verification. This scheme is more efficient than BGLS as BGLS requires O(n) pairing computation where n is the number of signers. However, many ID-based aggregate signature schemes [7, 9, 10] have been constructed from basic ID-based signature scheme.

The rest of the paper is organized as follows. In the next section, mathematical background of the proposed schemes has been explained. After that, the two proposed schemes have been presented in the next section. In Sect. 4, the efficiency analysis of the proposed ID-based aggregate signature schemes with other established ID-based aggregate signature schemes has been done.

2 Mathematical Background

Bilinear Pairing: Let G_1 be an additive cyclic group generated by P whose order is a prime q and G_2 be a multiplicative cyclic group of the same order q. A bilinear pairing is a map $e : G_1 \times G_1 \rightarrow G_2$ with the following properties:

(a) Bilinearity: $e(aP, bQ) = e(P; Q)^{ab}$ for all $P, Q \in G_1$ and all $b \in Z_q^*$.
(b) Non-degenerate: There exists $P, Q \in G_1$ such that $e(P, Q) \neq 1$.
(c) Computable: There is an efficient algorithm to compute (P, Q), for all $P, Q \in G_1$.

Additionally, the security of these proposed schemes depends on the hardness of the following Diffiee–Hellman problem.

Computational Diffie–Hellman Problem (CDHP): For $b \in_R Z_q^*$, given P, aP, bP, to compute abP is known as computational Diffie–Hellman problem which is a hard problem.

3 Two Proposed ID-Based Aggregate Signature Schemes

An aggregate signature scheme consists of six algorithms. They are **Setup, Extract, Sign, Verify, AggSign, and AggVerify**. The first four algorithms are for an ordinary identity-based signature scheme, and last two algorithms are for signature aggregation and aggregate signature verification. It works as follows. The first proposed aggregate signature scheme is presented in Sect. 3.1, and other one is presented in Sect. 3.2.

3.1 A Proposed ID-Based Aggregate Signature Schemes (First One)

SETUP: Given a security parameter k, the private key generator (PKG) runs the setup algorithm and outputs two groups G_1 of prime order q and G_2 of same order. The bilinear pairing is given as $e : G_1 \times G_2 \rightarrow G2$. PKG establishes the system parameters $q, G_1, G_2, P, Q, P_{\text{pub}}, P_{\text{pub}^2}, e, H_1, H_2$ where

1. P is the generator of group G_1.
2. PKG picks master key $s \in Z_p^*$ and computes $P_{\mathrm{pub}} = sP, P_{\mathrm{pub}^2} = s^2P$.
3. PKG also chooses two cryptographic hash functions, $H_1 : \{0,1\}^* \rightarrow G_1$ and $H_2 : \{0,1\}^* \rightarrow Z_q^*$.

EXTRACT: Let $P_1, P_2, \ldots, P_n$ denote all the users to join the signing process. The identity of P_i is denoted as ID_i. For user's identity ID_i, its public key $Q_{ID_i} = H_2(ID_i)$ and private key $S_{ID_i} = sQ_{ID_i}$. The user makes Q_{ID_i} public and keeps S_{ID_i} secret.

SIGN: For a message m_i, user with identity ID_i follows the steps below:

1. Choose a random number $r_i \in Z_q^*$, and broadcasts $U_i = r_iP$.
2. Calculate the value $h_i = H_2(m_i, ID_i, U_i)$
3. Calculate the value $V_i = r_iP + h_iS_{ID_i}$
4. The signature σ_i is then the pair (U_i, V_i).

VERIFY:

1. The designated player (DP) computes $U = \sum_{i=1}^{n} U_i$.
2. Compute $h_i = H_2(m_i, ID_i, U_i)$
3. Accept if $e(P_{\mathrm{pub}}, V_i) = e(U_i, P_{\mathrm{pub}})e\left(P_{\mathrm{pub}^2}, h_iQ_{ID_i}\right)$

AGGSIGN: DP computes $V = \sum_{i=1}^{n} V_i$. The aggregate signature on n different messages $m_1, m_2, \ldots, m_n$ given by n users $P_1, P_2, \ldots, P_n$ is $\sigma = (U, V)$.

AGGVERIFY: Given aggregate signature $\sigma = (U, V)$ by aggregating party and the list of $\langle ID, message \rangle$ pairs $\{ID_i, m_i\}$, the verifier verifies the aggregate signature compute

1. $$h_i = H_2(ID_i, m_i, U_i).$$
2. $$Q_{ID_i} = H_1(ID_i)$$
3. Accept the signature $\sigma = (U, V)$ if and only if

$$e(P_{\mathrm{pub}}, V) = e(P_{\mathrm{pub}}, U) \cdot e\left(P_{\mathrm{pub}^2}, \sum_{i=1}^{n} h_iQ_{ID_i}\right)$$

CORRECTNESS:

$$\begin{aligned} e(P_{\text{pub}}, V) &= e\left(P_{\text{pub}}, \sum_{i=1}^{n} V_i\right) \\ &= \prod_{i=1}^{n} e(P_{\text{pub}}, V_i) = \prod_{i=1}^{n} e(P_{\text{pub}}, r_iP + h_iS_{ID_i}) \\ &= e\left(P_{\text{pub}}, \sum_{i=1}^{n} (r_iP + h_iS_{ID_i})\right) = e\left(P_{\text{pub}}, \sum_{i=1}^{n}\left(r_iP + \sum_{i=1}^{n} h_iS_{ID_i}\right)\right) \\ &= e\left(P_{\text{pub}}, U + \sum_{i=1}^{n} h_iS_{ID_i}\right) = e(P_{\text{pub}}, U)e\left(P_{\text{pub}}, \sum_{i=1}^{n} h_iS_{ID_i}\right) \\ &= e(P_{\text{pub}}, U)e\left(s.P_{\text{pub}}, \sum_{i=1}^{n} h_iQ_{ID_i}\right) = e(P_{\text{pub}}, U)e\left(P_{\text{pub}^2}, \sum_{i=1}^{n} h_iQ_{ID^i}\right) \end{aligned}$$

3.2 Another Proposed Improved Identity-Based Aggregate Signature Scheme (Second One)

SETUP: Given a security parameter k, the private key generator (PKG) runs the setup algorithm and outputs two group G_1 of prime order q and G_2 of same order. The bilinear pairing is given as $e : G_1 \times G_1 \rightarrow G_2$. PKG establishes the system parameters $q, G_1, G_2, P, Q, P_{\text{pub}}, P_{\text{pub}^2}, e, H_1, H_2$ where

1. P and Q are the random generators of group G_1.
2. PKG picks master key $s \in Z_q^*$ and computes $P_{\text{pub}} = sP, P_{\text{pub}^2} = s^2P$.
3. PKG also chooses two cryptographic hash functions, $H_1 : \{0,1\}^* \rightarrow G_1$ and $H_2 : \{0,1\}^* \rightarrow Z_q^*$.

EXTRACT: Let $P_1, P_2, \ldots, P_n$ denote all the users to join the signing. The identity of P_i is denoted as ID_i. For user's identity ID_i, its public key $Q_{ID_i} = H_2(ID_i)$ and private key $s_{ID_i} = sQ_{ID_i}$. The user makes Q_{ID_i} public and keeps S_{ID_i} secret.

SIGN: For a message m_i, user with identity ID_i follows the steps below:

1. Choose a random number $r_i \in Z_q^*$ and broadcasts $U_i = r_iP_{\text{pub}}$
2. Calculate the value $h_i = H_2(m_i, ID_i, U_i)$
3. Calculate the value $V_i = r_iQ + h_iS_{ID_i}$
4. The signature σ_i is then the pair (U_i, V_i)

VERIFY:

1. The designated player computes $U = \sum_{i=1}^{n}$

2. Compute $h_i = H_2(m_i, ID_i, U_i)$
3. Accept if $e\left(P_{\text{pub}}, V_i\right) = e(U_i, Q)e\left(P_{\text{pub}^2}, h_i Q_{ID_i}\right)$

AGGSIGN: DP computes $V = \sum_{i=1}^{n} V_i$. The aggregate signature on n different messages $m_1, m_2, \ldots, m_n$ given by n users $P_1, P_2, \ldots, P_n$ is $\sigma = (U, V)$

AGGVERIFY: Given aggregate signature $\sigma = (U, V)$ by aggregating party and the list of $\langle ID, message \rangle$ pairs $\{ID_i, m_i\}$, the verifier verifies the signature by computing the following:

1. $h_i = H_2(ID_i, m_i, U)$
2. Accept the signature $\sigma = (U, V)$ if and only if

$$e\left(P_{\text{pub}}, V\right) = e(Q, U).e\left(P_{\text{pub}^2}, \sum_{i=1}^{n} h_i Q_{ID_i}\right)$$

- **CORRECTNESS**:

$$\begin{aligned}
e\left(P_{\text{pub}}, V\right) &= e\left(P_{\text{pub}}, \sum_{i=1}^{n} V_i\right) \\
&= \prod_{i=1}^{n} e(P_{\text{pub}}, V_i) \\
&= \prod_{i=1}^{n} e(P_{\text{pub}}, r_i Q + h_i S_{ID_i}) \\
&= \prod_{i=1}^{n} e(r_i P_{\text{pub}}, Q) e\left(s P_{\text{pub}}, h_i Q_{ID_i}\right) \\
&= \prod_{i=1}^{n} e(U_i, Q) e(P_{\text{pub}^2}, h_i Q_{ID_i}) \\
&= e\left(Q, \sum_{i=1}^{n} U_i\right) e\left(P_{\text{pub}^2,} \sum_{i=1}^{n} h_i Q_{ID_i}\right) \\
&= e(Q, U) e\left(P_{\text{pub}^2}, \sum_{i=1}^{n} h_i Q_{ID_i}\right)
\end{aligned}$$

4 Efficiency Comparison

In this section, we will compare our schemes with the schemes in Refs. [7, 9, 10] as we have constructed these two schemes from the idea achieved from those papers. In general, the number of pairing computations of identity-based aggregate signature schemes (IBASs) is proportional to that of signers. But, our proposed IBAS schemes require constant number of pairing computations in aggregated signature

Table 1 Computational complexity of IBAS schemes in the number n of signers

IBAS scheme	Aggregated signature length	Individual sign	Individual signature verify	Aggregate sign	Aggregate signature verify
Ref. [9]	$2\|G_1\|$	$3\Delta_{SM} + (n)\Delta_{PA} + 1\Delta_{\mathrm{Hash}}$	$(n-1)\Delta_{PA} + 1\Delta_{\mathrm{Hash}} + 3\Delta_{PO}$	$(n-1)\Delta_{PA}$	$2\Delta_{\mathrm{Hash}} + 3\Delta_{PO} + n\Delta_{SM} + (n-1)\Delta_{PA}$
Ref. [7]	$(n+1)\|G_1\|$	$3\Delta_{SM} + 1\Delta_{PA} + 1\Delta_{\mathrm{Hash}}$	$1\Delta_{\mathrm{Hash}} + 3\Delta_{PO}$	$(n-1)\Delta_{PA}$	$2\Delta_{\mathrm{Hash}} + 3\Delta_{PO} + n\Delta_{SM} + 2(n-1)\Delta_{PA}$
Ref [10]	$(n+1)\|G_1\|$	$3\Delta_{SM} + 1\Delta_{PA} + 1\Delta_{\mathrm{Hash}}$	$1\Delta_{PA} + 1\Delta_{SM} + 2\Delta_{\mathrm{Hash}} + 2\Delta_{PO}$	$(n-1)\Delta_{PA}$	$2\Delta_{\mathrm{Hash}} + 2\Delta_{PO} + n\Delta_{SM} + \{n + (n-1)\}\Delta_{PA}$
First proposed scheme	$2\|G_1\|$	$2\Delta_{SM} + 1\Delta_{PA} + 1\Delta_{\mathrm{Hash}}$	$(n-1)\Delta_{PA} + 1\Delta_{\mathrm{Hash}} + 3\Delta_{PO}$	$(n-1)\Delta_{PA}$	$2\Delta_{\mathrm{Hash}} + 3\Delta_{PO} + n\Delta_{SM} + (n-1)\Delta_{PA}$
Second proposed scheme	$2\|G_1\|$	$3\Delta_{SM} + 1\Delta_{PA} + 1\Delta_{\mathrm{Hash}}$	$(n-1)\Delta_{PA} + 1\Delta_{\mathrm{Hash}} + 3\Delta_{PO}$	$(n-1)\Delta_{PA}$	$2\Delta_{\mathrm{Hash}} + 3\Delta_{PO} + n\Delta_{SM} + (n-1)\Delta_{PA}$

verification process and are independent of the number of signers. An efficiency comparison of our schemes with the existing established schemes is given in Table 1. Here, Δ_{PO}, Δ_{PA}, Δ_{Hash}, and Δ_{SM} denote the number of pairing operations, point addition in G_1 group, hash function, and scalar multiplications in G_1 group, respectively.

5 Conclusion

In this paper, we propose two ID-based aggregate signature schemes with constant pairings needed in signature verification process. We observe that the first scheme is as same efficient as the scheme [10] which assumed to be the most efficient IBAS scheme until now. The security of the scheme is purely based on difficulty of solving computational Diffie–Hellman problem in the random oracle model. Due to page limitation, the security proof is not given in the paper. Just like all other pairing-based cryptosystems, it is not only simple and efficient but also has a shorter signature size.

References

1. Boneh, Dan and Gentry, Craig and Lynn, Ben and Shacham, Hovav: Aggregate and verifiably encrypted signatures from bilinear maps, Advances in cryptology EUROCRYPT, Springer, 416–432 (2003).
2. Boneh, Dan and Lynn, Ben and Shacham, Hovav: Short signatures from the Weil pairing, Journal of Cryptology, Springer, 17, 297–319 (2004).
3. Meffert, Dennis: Bilinear pairings in cryptography, Masters thesis, Radboud Universiteit Nijmegen (2009).
4. Shakerian, Reza and Pour, Touraj Mohammad and Kamali, Seyed Hossein and Hedayati, Maysam: An identity based public key cryptography blind signature scheme from bilinear pairings, 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT), IEEE 7, 28–32 (2010).
5. Shamir, Adi: Identity-based cryptosystems and signature schemes, Workshop on the Theory and Application of Cryptographic Techniques, Springer, 47–53 (1984).
6. Cheon, Jung Hee and Kim, Yongdae and Yoon, HyoJin and others: A New ID-based Signature with Batch Verification, IACR Cryptology ePrint Archive, 131 (2004).
7. Wang, Zhu and Wu, Qian and Ye, Ding-feng and Chen, Hui-yan: Practical identity based aggregate signature from bilinear maps, Journal of Shanghai Jiaotong University (Science), Springer 13, 684–687 (2008).
8. Wen, Yiling and Ma, Jianfeng: An aggregate signature scheme with constant pairing operations, International Conference on Computer Science and Software Engineering IEEE, 3, 830–833 (2008).
9. Yu, Yike and Zheng, Xuefeng and Sun, Hua: A new ID-based aggregate signature with constant pairing operations, Second International Conference on Networks Security Wireless Communications and Trusted Computing (NSWCTC), IEEE 2, 188–191 (2010).
10. Shim, Kyung-Ah: An ID-based aggregate signature scheme with constant pairing computations, Journal of Systems and Software, Elsevier 83, 1873–1880 (2010).