Insecurity of One Decentralized Attribute-based Signature Scheme for Social Co-governance

Zhengjun Cao, Lihua Liu

Abstract. We show that the attribute-based signature scheme [Information Sciences, 654(2024), 119839] is insecure, because an adversary can generate valid signatures for any message even though he cannot access the signer's secret key. The four components of signature $\{\delta_1, \delta_2, \delta_3, \delta_4\}$ are not tightly bound to the target message M and the signer's public key. The dependency between the signer's public key and secret key is not properly used to construct any intractable problem. The inherent flaw results in that the adversary can find an efficient signing algorithm functionally equivalent to the valid signing algorithm.
Keywords: Attribute-based signature, forgery attack, signing algorithm, verification algorithm, anonymity.

1 Introduction

Digital signature can provide a means for an entity to bind its identity to a piece of information. The process of signing entails transforming the message and some secret information held by the entity into a tag called a signature [8]. A verification algorithm is a method for verifying that a digital signature is authentic, i.e., was indeed created by the specified entity. Attribute-based signature (ABS) allows a party, who possesses a set of attributes from the authority, to sign a message with fine-grained control over identifying information. The signature reveals no more than the fact that a single user with some set of attributes satisfying the predicate has attested to the message [7]. Okamoto and Takashima [9, 10] discussed some decentralized ABS schemes for non-monotone predicates in the standard model. Rao and Dutta [12] designed an ABS scheme which realized expressive access structures. Sakai et al. [13] developed an efficient ABS for circuits using bilinear maps. Datta et al. [3] proposed an ABS scheme for unbounded arithmetic branching programs.

Perera et al. [11] presented a full anonymous attribute-based group signature with verifier-local revocation and member registration. Chen et al. [2] designed an efficient attribute based server-aided verification signature by using the attribute tree as access policy that expresses flexible access control. In 2023, Kang et al. [5] presented a traceable and forward-secure ABS scheme with constant-size, in order to alleviate the damage induced by key exposure and trace the real identity of signer by attribute authority when the signer occurs abusing behavior. Kang

Z. Cao, Department of Mathematics, Shanghai University, Shanghai, 200444, China.
L. Liu, Department of Mathematics, Shanghai Maritime University, Shanghai, 201306, China.
Email: liulh@shmtu.edu.cn

et al. [6] proposed an outsourced ABS scheme with constant signature length unrelated to the number of required attributes. Very recently, Delerablée et al. [4] have studied the problems of ABS with advanced delegation and tracing.

In 2024, Tao, Cui, and Iftekhar [14] have presented a decentralized ABS scheme. Though the Tao-Cui-Iftekhar signature scheme is interesting, we find it is insecure. An adversary can find an efficient signing algorithm functionally equivalent to the valid signing algorithm, even though he cannot compute the private key information of any signer. This drawback is due to that the four components of signature $\{\delta_1, \delta_2, \delta_3, \delta_4\}$ are not tightly bound to the target message M and the signer's public key. We also clarify some misunderstandings in the signature scheme.

2 Review of Tao-Cui-Iftekhar signature scheme

In the considered scenario, there are four entities. The Key Generate Center (KGC) consists of Attribute Authorities (AAs), who grants users attributes, initializes the system parameters and issues private key for users. The Cloud Service Provider (CSP) is responsible for generating partial signatures. Data owners share their information on demand. Data users can access shared data. The scheme consists of six phases. For readers' convenience, we now briefly describe the scheme [14] as follows.

$\diamond$ **Setup**. In this phase, the KGC initializes the system parameters. Choose groups G and G_T of order $N = pq$ (see page 6, Ref.[14]), where p and q are two big primes. $g \in G$ is a generator. $e : G \times G \to G_T$ is a bilinear map. The attributes set U are managed by $AA_i, i = 1, \cdots, n$. Each AA_i manages a subset $U_i \subset U$. Let $\overline{\boldsymbol{W}} = \{j_1, \cdots, j_{d-1}\}$ be the set of default attributes. Select hash functions $H, H_2 : \{0,1\}^* \to G$. Pick $\tau \in Z_p^*$ to compute $T = g^\tau$. For each AA_i, pick $x_i \in Z_p^*$ to compute $P_i = g^{x_i}$. For each attribute, select $t_{i,j} \in Z_p^*$. The system public key is set as

$$PK = \{H, H_2, e, g, T, P_1, \cdots, P_n, \overline{\boldsymbol{W}}\}.$$

The system master secret key is set as

$$MSK =< x_1 \cdots, x_n, \{t_{i,j}\}_{i=1,\cdots,n, j \in U_j} > .$$

$\diamond$ **KeyGeneration**. In this phase, the authority AA_i generates the private key for each user. Let $H_1(\cdot)$ be a pseudo-random function. User ID_k possesses an attribute set U_{ID_k}. The AA_i picks $\xi \in Z_p^*$ to compute

$$\Gamma_{ID_{i,k}|k=1,\cdots,n} = H_1(ID_k + \xi),\ D_{0,i|i\in Group} = g^{\Gamma_{ID_{i,k}}},\ D_{1,i|i\in Group} = D_{0,i} g^{-x_i}.$$

Compute $D_{2,i,j|i\in Group} = D_{0,i} H(j)^{t_{i,j}}$, $j \in (U_{ID_k} \cup \overline{\boldsymbol{W}})$. The private key of user ID_k is finally set as $SK =< \{D_{0,i}, D_{1,i}, D_{2,i,j}\}_{i=1,\cdots,n, j\in(U_{ID_k}\cup\overline{\boldsymbol{W}})} >$. We refer to the section §3.2 in Ref.[14] for the details.

$\diamond$ **Outsourced-signing**. In this phase, the CSP generates the partial signature for a target message. Given a message M and the access control policy U'_{ID_k}, the signer ID_k defines the access control policy U'_{ID_k}, where $\mid U'_{ID_k} \mid = k$. The CSP selects $d - k$ attributes set $W \subset \overline{\boldsymbol{W}}$. Let $\gamma_{(\mathcal{S},\rho)} = U'_{ID_k} \cup W$. Construct the access matrix $\boldsymbol{S}_{l*m}$ with the injective map ρ. Pick $b \in Z_p^*$, and take the vector $v = \{b, v_1, v_2, \cdots, v_{m-1}\} \in \{Z_p^*\}^m$. Define $\lambda_i = \mathcal{S}_i \cdot v$ where S_i is the ith

row of $\boldsymbol{S}$. Choose $\omega_i \in Z_P^*$ such that $\sum_{x\in\rho(\mathcal{S}_i)}(\omega_i * \mathcal{S}_i) = 1$. For each attribute $x \in \gamma(\boldsymbol{S}, \rho) \cup W$, choose $r_{i,j} \in Z_p^*$. If $x \in \rho(S_i)$, compute $\delta'_{1,x} = g^{r_{i,j}}(D_{2,i,j}T)^{\lambda_i}$. If $x \in W$, compute $\delta'_{1,x} = g^{r_{i,j}}$. Then compute

$$\delta'_2 = H_2(M) \prod_{x\in\rho(S_i)} (\delta'_{1,x})^{\omega_i} \prod_{i\in W} (P_i D_{1,i} \prod_{j\in W} \delta'_{1,x}),$$
$$\delta'_3 = \prod_{x\in\rho(S_i)} (g^{r_{i,j}} H(j)^{t_{i,j}})^{\omega_i} \prod_{x\in W} (\delta'_{1,x}), \quad \delta'_4 = T^b, \quad \delta'_5 = D_{0,i}^b \prod_{i\in W} D_{0,i}.$$

the CSP computes the partial signature $\{\delta'_2, \delta'_3, \delta'_4, \delta'_5\}$.

$\diamond$ **Signing.** The algorithm is run by data owner ID_k. Given M and $\{\delta'_2, \delta'_3, \delta'_4, \delta'_5\}$, the signer picks $\alpha, \gamma \in Z_p^*$ to compute

$$\delta_1 = \delta'_3, \quad \delta_2 = g^\alpha \delta'_2, \quad \delta_3 = H_2(M)^\gamma g^\alpha \delta'_5, \quad \delta_4 = e(g, \delta'_4 H_2(M)^{-\gamma}).$$

Output the final signature $\{\delta_1, \delta_2, \delta_3, \delta_4\}$.

$\diamond$ **Verification.** The algorithm is run by any data user. Given M and $\{\delta_1, \delta_2, \delta_3, \delta_4\}$, check that

$$e(g, \delta_2) = e(g, \delta_1) * e(g, H_2(M)) * e(g, \delta_3) * \delta_4 \tag{1}$$

If true, accept the signature. Otherwise, reject it.

$\diamond$ **Batch-Verification.** For n messages M^{ID_i} and signatures δ^{ID_i}, $i = 1, \cdots, n$, the verifier checks that

$$e\left(g, \prod_{i=1}^n \delta_2^{ID_i}\right) = e\left(g, \prod_{i=1}^n \delta_1^{ID_i}\right) * e\left(g, \prod_{i=1}^n H_2(M^{ID_i})\right) * e\left(g, \prod_{i=1}^n \delta_3^{ID_i}\right) * \prod_{i=1}^n \delta_4^{ID_i}$$

If true, accept the signatures. Otherwise, reject.

3 Security analysis of Tao-Cui-Iftekhar signature scheme

3.1 Universal forgery against the signature scheme

For a signature scheme, the goal of an adversary is to forge signatures—produce signatures which will be accepted as those of some other entity. There are three kinds of forgeries [8]: universal forgery (total break), selective forgery, and existential forgery. For the universal forgery, an adversary can either compute the private key of the signer, or find an efficient signing algorithm equivalent to the valid signing algorithm. There are two types of attacks against public-key signature schemes: key-only attack, in which the adversary knows only the signer's public key; and message attack, in which the adversary can access the signer's public key, besides he can examine signatures corresponding to either known or chosen messages.

We find the Tao-Cui-Iftekhar signature scheme is insecure, because an adversary can generate valid signatures for any message M. The adversary can find a signing algorithm equivalent to the original signing algorithm even though he cannot access the singer's private key.

For example, the adversary picks three elements $\delta_1, \delta_2, \delta_3 \in G$ and computes

$$\delta_4 = \frac{e(g, \delta_2)}{e(g, \delta_1) * e(g, H_2(M)) * e(g, \delta_3)} \tag{2}$$

where $g \in G$, and e is a bilinear map, both are system public parameters and accessible to any adversary. Apparently, the forged signature $\{\delta_1, \delta_2, \delta_3, \delta_4\}$ and the message M can pass the verification Eq.(1).

The drawback is due to that the four components of signature $\delta = \{\delta_1, \delta_2, \delta_3, \delta_4\}$ are simply used for the verification, not truly bound to the target message M and any entity's public key. The dependency between the signer's public key and secret key is not properly used to construct any intractable problem, such as Elliptic Curve Discrete Logarithm Problem (ECDLP), and Computational Diffie-Hellman Problem (CDHP).

3.2 Unbalance between anonymity and forgeability

Unconditional full anonymity (UFA) requires not only that the scheme satisfies unconditional weak anonymity, but also that the adversary cannot get any user's effective information from signature. In order to achieve UFA, Tao, Cui, and Iftekhar [14] proposed the decentralized ABS scheme.

The scheme is really anonymous because the four components of signature $\{\delta_1, \delta_2, \delta_3, \delta_4\}$ are independent of the signer's identity ID_k. An adversary cannot retrieve the identity from the signature. Besides, the verification equation $e(g, \delta_2) = e(g, \delta_1) * e(g, H_2(M)) * e(g, \delta_3) * \delta_4$ involves only the system's public parameter g, hash function $H_2(\cdot)$, and the signed message M. Clearly, the adversary cannot get any signer's effective information from the data. But the scheme fails to keep the balance between unforgeability and anonymity. The four components $\delta_1, \delta_2, \delta_3, \delta_4$ are linearly invoked. More seriously, the signer's private key SK is not truly invoked in the signing phase. Actually, the signer ID_k is only required to pick two random numbers $\alpha, \gamma \in Z_p^*$ so as to transform the partial signature $\{\delta_2', \delta_3', \delta_4', \delta_5'\}$ into a final signature $\{\delta_1, \delta_2, \delta_3, \delta_4\}$. The fatal flaw results in the universal forgery.

3.3 Some misunderstandings

The scheme requires that G and G_T are two multiplicative groups of composite order $N = pq$. But the property is never used in the later description. Bilinear groups of composite order were introduced by Boneh [1] for designing homomorphic public key encryptions. The operations in such groups are somewhat inefficient. The setting is not compatible with the lightweight property. So, it could specify that "G and G_T are two multiplicative groups of prime order p".

The scheme specifies two hash functions $H, H_2 : \{0,1\}^* \to G$. Both have the same domain and codomain. In this case, it only needs to specify one hash function.

3.4 Further discussions

It seems difficult to improve the scheme without a thorough design methodology. For conveniences, we refer to Ref.[11] for an anonymous attribute-based group signature scheme, which achieves either unforgeability or anonymity, and can be treated as a replacement for some cases.

Even though we tried but failed to present an improvement of the Tao-Cui-Iftekhar signature, We realize that it is urgent to point out the flaw so as to remind readers of the misuse of this protocol. As we see, proving and disproving a cryptographic protocol are just two sides of one coin. Without intensive and on-going security inspections, it is impossible to turn a literal protocol into a practical protocol.

4 Conclusion

We show that the Tao-Cui-Iftekhar attribute-based signature scheme is insecure against universal forgery attack. It fails to keep the balance between unforgeability and anonymity. We notice that it seems difficult to fix the scheme without a new design methodology. We also refer to an anonymous attribute-based group signature, which could be taken as a possible replacement for some cases. The findings in this note could be helpful for the future works on designing attribute-based signature schemes.

References

[1] D. Boneh. Bilinear groups of composite order. In T. Takagi, T. Okamoto, E. Okamoto, and T. Okamoto, editors, *Proceedings of Pairing-Based Cryptography - Pairing 2007, First International Conference, Tokyo, Japan, July 2-4, 2007*, volume 4575 of *Lecture Notes in Computer Science*, page 1. Springer, 2007.

[2] Y. Chen, J. Li, C. Liu, J. Han, Y. Zhang, and P. Yi. Efficient attribute based server-aided verification signature. *IEEE Trans. Serv. Comput.*, 15(6):3224–3232, 2022.

[3] P. Datta, T. Okamoto, and K. Takashima. Efficient attribute-based signatures for unbounded arithmetic branching programs. In D. Lin and K. Sako, editors, *Proceedings of Public-Key Cryptography - PKC 2019 - 22nd IACR International Conference on Practice and Theory of Public-Key Cryptography, Beijing, China, April 14-17, 2019, Part I*, volume 11442 of *Lecture Notes in Computer Science*, pages 127–158. Springer, 2019.

[4] C. Delerablée, L. Gouriou, and D. Pointcheval. Attribute-based signatures with advanced delegation, and tracing. In E. Oswald, editor, *Proceedings of Topics in Cryptology - CT-RSA 2024 - Cryptographers' Track at the RSA Conference 2024, San Francisco, CA, USA, May 6-9, 2024*, volume 14643 of *Lecture Notes in Computer Science*, pages 224–248. Springer, 2024.

[5] Z. Kang, J. Li, J. Shen, J. Han, Y. Zuo, and Y. Zhang. TFS-ABS: traceable and forward-secure attribute-based signature scheme with constant-size. *IEEE Trans. Knowl. Data Eng.*, 35(9):9514–9530, 2023.

[6] Z. Kang, J. Li, Y. Zuo, Y. Zhang, and J. Han. OABS: efficient outsourced attribute-based signature scheme with constant size. *IEEE Internet Things J.*, 11(23):38167–38177, 2024.

[7] H. K. Maji, M. Prabhakaran, and M. Rosulek. Attribute-based signatures. In A. Kiayias, editor, *Proceedings of Topics in Cryptology - CT-RSA 2011 - The Cryptographers' Track at the RSA Conference 2011, San Francisco, CA, USA, February 14-18, 2011.*, volume 6558 of *Lecture Notes in Computer Science*, pages 376–392. Springer, 2011.

[8] A. Menezes, P. Oorschot, and S. Vanstone. *Handbook of Applied Cryptography.* CRC Press, USA, 1996.

[9] T. Okamoto and K. Takashima. Decentralized attribute-based signatures. In K. Kurosawa and G. Hanaoka, editors, *Proceedings of Public-Key Cryptography - PKC 2013 - 16th International Conference on Practice and Theory in Public-Key Cryptography, Nara, Japan, February 26 - March 1, 2013.*, volume 7778 of *Lecture Notes in Computer Science*, pages 125–142. Springer, 2013.

[10] T. Okamoto and K. Takashima. Efficient attribute-based signatures for non-monotone predicates in the standard model. *IEEE Trans. Cloud Comput.*, 2(4):409–421, 2014.

[11] M. N. S. Perera, T. Nakamura, M. Hashimoto, H. Yokoyama, and K. Sakurai. Almost fully anonymous attribute-based group signatures with verifier-local revocation and member registration from lattice assumptions. *Theor. Comput. Sci.*, 891:131–148, 2021.

[12] Y. S. Rao and R. Dutta. Efficient attribute-based signature and signcryption realizing expressive access structures. *Int. J. Inf. Sec.*, 15(1):81–109, 2016.

[13] Y. Sakai, N. Attrapadung, and G. Hanaoka. Practical attribute-based signature schemes for circuits from bilinear map. *IET Inf. Secur.*, 12(3):184–193, 2018.

[14] Q. Tao, X. Cui, and A. Iftekhar. A novel lightweight decentralized attribute-based signature scheme for social co-governance. *Inf. Sci.*, 654:119839, 2024.