Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 761487, 5 pages
http://dx.doi.org/10.1155/2014/761487

Research Article

On Delegatability of Some Strong Designated Verifier Signature Schemes

Baoyuan Kang, Hao Xu, and Yongzheng Niu

School of Computer Science and Software, Tianjin Polytechnic University, No. 399, Binshuixi Road, Tianjin 300387, China

Correspondence should be addressed to Baoyuan Kang; baoyuankang@aliyun.com

Received 25 December 2013; Revised 27 February 2014; Accepted 7 March 2014; Published 1 April 2014

Academic Editor: Kwok-Wo Wong

A strong designated verifier signature scheme makes it possible for a signer to convince a designated verifier that she has signed a message in such a way that the designated verifier cannot transfer the signature to a third party, and no third party can even verify the validity of a designated verifier signature. In 2005, Lipmaa, Wang, and Bao identified a new essential security property, non delegatability, of designated verifier signature schemes. Briefly, in a non delegatability designated verifier signature scheme, neither a signer nor a designated verifier can delegate the signing rights to any third party without revealing their secret keys. However, this paper shows that four recently proposed strong designated verifier signature schemes are delegatable. These schemes do not satisfy non delegatability secure requirement of strong designated verifier signature schemes.

1. Introduction

Ensuring the integrity and the authenticity of the origin of a message is one of the goals of cryptography, and standard authentication tools are digital signatures. Digital signature schemes allow a receiver of a signature, Bob, to verify that the signature received is indeed sent by the sender, Alice. And Bob can convince any third party that Alice has indeed sent him the message. This is also referred to as nonrepudiation in the sense that Alice cannot deny the fact that she has sent a signature to Bob. Nonrepudiation is a very useful property for the authenticity of the origin of a message when dispute could occur at some later time. On the other hand, in numerous applications such as tender, electronic voting, or electronic auctions, the public verification and nonrepudiation properties of a signature are not desired. Let us consider the following example [1].

Suppose that a public institution initiates a call for tenders, asking some companies to propose their prices for a set of instruments and tasks to be accomplished. The institution may require the companies to sign their offers in order to make sure that they are actually authentic and originated from whom they claim to be. This is a valid requirement, but no company involved in this process desires its offer to affect other tenders' decisions. That is, a company may capture a competitor's signed offer on the transmission line (to the institution) and prepares its offer consequently in order to increase its chance to be selected by the institution. The here raised question is about the conflict between authenticity and privacy.

To satisfy the above requirements in signature schemes, Jakobsson et al. [2] firstly proposed the concept of strong designated verifier signatures (SDVS). A SDVS scheme is special type of digital signature which provides message authentication without nonrepudiation. In a SDVS scheme, suppose Alice, the signer, has sent a signature to Bob, the designated verifier. Bob can use his private key to verify the validity of the signature. But Bob cannot prove to a third party that Alice has created the signature. Since Bob can efficiently simulate signatures that are indistinguishable from Alice's signature. The SDVS fit into various cryptographic applications such as privacy preserving cloud computing [3] and social networks [4]. They also are useful in some new fields, such as cognitive computing [5], where a brainy robot needs to authenticate its owner and keeps no evidences of its owner's authentication.

After Saeednia et al. [1] formalized the notion of SDVS in 2003, many SDVS schemes have been proposed [6–18]. Based on nondelegatability proposed by Lipmaa, Wang, and

Bao in 2005, an essential security property of designated verifier signature schemes, Huang et al. [14] proposed a security model for SDVS scheme. The model is stricter than the previous one [11]. All schemes [6–13] are insecure in Huang et al.'s model. In a nondelegatability designated verifier signature scheme, neither a signer nor a designated verifier can delegate the signing rights to any third party without revealing their secret keys. Recently, four strong designated verifier signature schemes are proposed [15–18]. However, in this work, we show that the four schemes are delegatable. So, they are insecure.

The remainder of this paper is organized as follows. Some basic concepts are introduced in Section 2. In Section 3, we review four designated verifier signature schemes and present delegation attacks on them. Finally, Section 4 concludes the paper.

2. Preliminaries

In this section, we briefly review the basic concepts of bilinear pairings and model of strong designated verifier signatures.

2.1. Basic Concepts on Bilinear Pairings. Let G_1 be a cyclic additive group and G_2 a cyclic multiplicative group of the same order q. An admissible bilinear pairing is a map $e : G_1 \times G_1 \to G_2$, which satisfies the following properties.

(i) *Bilinearity.* One has $e(aP, bQ) = e(P, Q)^{ab}$ for all $P, Q \in G_1$, $a, b \in Z_q^*$. This can also be stated as $e(P + Q, R) = e(P, R)e(Q, R)$ and $e(P, Q + R) = e(P, Q)e(P, R)$ for all $P, Q, R \in G_1$.

(ii) *Nondegeneracy.* There exists $P, Q \in G_1$, such that $e(P, Q) \neq 1$.

(iii) *Computability*. There is an efficient algorithm to compute $e(P, Q)$ for all $P, Q \in G_1$.

2.2. Complexity Assumptions

Definition 1 (bilinear Diffie-Hellman (BDH) problem). Given randomly chosen $P \in G_1$, as well as aP, bP, cP (for unknown randomly chosen $a, b, c \in Z_q^*$), compute $e(P, P)^{abc}$.

Definition 2 (BDH assumption). The BDH assumption (t, ε) holds in the bilinear setting (G, G_2, e, q, P), if there is no probabilistic polynomial-time adversary A that runs in time at most t and $\Pr[a, b, c \in Z_q^* : e(P, P)^{abc} \leftarrow A(aP, bP, cP)] > \varepsilon$.

2.3. Model of Strong Designated Verifier Signature Scheme. Here, we introduce the concept of strong designated verifier signature in identity-based setting. An identity-based strong designated verifier signature scheme (IDSDVS) consists of five algorithms (that may be randomized) as follows.

(i) Parameter Generation (*Setup*) is an algorithm that accepts a security parameter k and outputs a string consisting of system parameters and master key.

(ii) Key Extraction (*Extract*) is an algorithm that accepts system parameters and master key and an arbitrary string $\mathrm{ID} \in \{0, 1\}^*$ outputs a private key U_{ID}. Here ID is the user's identity and will be used as the user's public key.

(iii) Signature Generation (*Sign*) is an algorithm that accepts system parameters, the signer's private key U_{ID}, a message m, and the designated verifier's public key ID_V and outputs the signature δ on the message m.

(iv) Designated Verification (*Ver*) is an algorithm that accepts system parameters, the signer's identity ID_S, a message m, the designated verifier's public key ID_V, and private key u_V and the signature δ on the message m outputs either accept or reject as the verification decision.

(v) Transcript Simulation is the algorithm that the designated verifier runs to produce identically distributed transcripts which are indistinguishable from the signature produced by the signer.

The IDSDVS scheme should satisfy the following security properties.

(i) *Correctness.* A properly formed IDSDVS must be accepted by the verifying algorithm.

(ii) *Nontransferability.* We require an IDSDVS scheme to be nontransferable. The nontransferability property is ensured by a transcript simulation algorithm that can be performed by all designated verifiers to produce an indistinguishable signature from the one that should be produced by the signature holder.

(iii) *Unforgeability.* It is computationally infeasible to construct a valid IDSDVS signature without the knowledge of the private key of either the signer or the designated verifier.

(iv) *Nondelegatability.* It requires an adversary to "know" a secret key of a signer or a designated verifier if the adversary can produce a valid signature on a message.

3. Four Designated Verifier Signature Schemes and Attacks on Them

3.1. Lee et al.'s Scheme. Lee et al.'s scheme [16] can be described as follows.

Let p and q be two large primes such that $q \mid p - 1$ and g an element of Z_p^* of order q. The message to be signed is $m \in Z_p$. Let signer Alice's public key be $y_A = g^{x_A} \bmod p$, where $x_A \in Z_q^*$ is her secret key, and designated verifier Bob's public key be $y_B = g^{x_B} \bmod p$, where $x_B \in Z_q^*$ is his secret key. One-way hash function H outputs values in Z_q. Suppose that Alice wants to send a strong designated verifier signature (t, c, r, s) with a message m to Bob.

(i) *Signature generation*. Alice chooses two random numbers k_1 from Z_q^* and k_2 from Z_q and generates a signature (t, c, r, s) as follows:

$$\begin{aligned} t &= g^{k_1} \bmod p, \\ c &= m y_B^{k_2} \bmod p, \\ r &= H\left(m, g^{k_2}\right), \\ s &= k_1^{-1}\left(x_A r - k_2\right) \bmod q. \end{aligned} \tag{1}$$

(ii) *Message Recovery and Verification*. Upon receiving (t, c, r, s) from Alice, Bob recovers the message and verifies the signature by computing

$$\begin{aligned} m &= c\left(t^s y_A^{-r}\right)^{x_B} \bmod p, \\ r &\stackrel{?}{=} H\left(m, y_A^r t^{-s}\right). \end{aligned} \tag{2}$$

(iii) *Transcript Simulation*. Bob can simulate the designated verifier signature (t, c, r, s) of m. Bob selects two random values $w_1 \in Z_q^*$ and $w_2 \in Z_q$. Then he computes (t, c, r, s) as follows:

$$\begin{aligned} t &= y_A^{w_1^{-1}} \bmod p, \\ c &= m y_A^{x_B w_1^{-1} w_2} \bmod p, \\ r &= H\left(m, y_A^{w_1^{-1} w_2}\right) \\ s &= w_1 r - w_2 \bmod q. \end{aligned} \tag{3}$$

Attack on Lee et al.'s Scheme. Assume that the signer discloses $D = y_B^{x_A}$ or the designated verifier discloses $y_A^{x_B} (= y_B^{x_A})$ to any third party T. Given any message m^*, T selects two random values $d_1 \in Z_q^*$ and $d_2 \in Z_q$. Then he computes (t^*, c^*, r^*, s^*) as follows:

$$\begin{aligned} t^* &= y_A^{d_1^{-1}} \bmod p, \\ c^* &= m^* D^{d_1^{-1} d_2} \bmod p, \\ r^* &= H\left(m, y_A^{d_1^{-1} d_2}\right), \\ s^* &= d_1 r^* - d_2 \bmod q. \end{aligned} \tag{4}$$

T generates a simulated signature (t^*, c^*, r^*, s^*). Bob verifies whether $r^* = H(m, y_A^{r^*} t^{*-s^*})$ and recovers message $m^* = c^*(t^{*s^*} y_A^{-r^*})^{x_B} \bmod p$. The verification accepts since

$$\begin{aligned} H\left(m^*, y_A^{r^*} t^{*-s^*}\right) &= H\left(m^*, y_A^{r^*}\left(y_A^{d_1^{-1}}\right)^{-(d_1 r^* - d_2)}\right) \\ &= H\left(m^*, y_A^{d_1^{-1} d_2}\right) = r^*. \end{aligned} \tag{5}$$

Therefore, Lee et al.'s scheme is delegatable.

3.2. Yang et al.'s Scheme. Yang et al.'s certificateless strong designated verifier signature scheme [18] consists of the following six algorithms.

(i) *Setup*. Given a security parameter l, a KGC chooses two groups G_1 and G_2 of the same prime order $q > 2^l$ and a modified Tate pairing map $e: G_1 \times G_1 \to G_2$. P is a generator of group G_1; then the KGC selects two distinct cryptographic hash functions $H_1: \{0,1\}^* \to G_1$, $H_2: \{0,1\}^* \times G_1^3 \to Z_q^*$, picks a random $s \in Z_q^*$ as the master key, computes the system public key $P_0 = sP$, and publishes params $:= \{l, G_1, G_2, e, q, P, P_0, H_1, H_2\}$ but keeps s secret.

(ii) *Partial-Private-Key-Extract*. Given an identity $\mathrm{ID}_i \in (0,1)^*$, $i \in \{A, B\}$, this paper assumes that user A is the signer and B is the designated verifier, the KGC computes $Q_i = H_1(\mathrm{ID}_i)$, $d_i = sQ_i$, and sends d_i to a user with identity ID_i as his partial private key by a secure channel.

(iii) *User-Key-Extract*. On inputs *params* and the user's identity ID_i ($i \in \{A, B\}$), the algorithm picks a random $x_i \in Z_q^*$ as the user's secret value and computes $pk_i = x_i P$ as his public key.

(iv) *CLSDVS-Sign*. On inputs *params*, signer A's identity ID_A, his private key pair (d_A, x_A), and a message $m \in (0,1)^*$, the algorithm works as follows.

(1) Pick a random value $r \in Z_q^*$ and compute $U = rP$.
(2) Compute $h = H_2(m, U, pk_A, x_A pk_B) \in Z_q^*$.
(3) Compute $V = rP_0 + hd_A$ and $T = e(V, Q_B)$.

The signature on message m is $\sigma = (U, T)$.

(v) *CLSDVS-Verify*. To verify a signature σ on a message m for an identity ID_A with public key pk_A, the designated verifier B acts as follows.

(1) Parse $\sigma = (U, T)$.
(2) Compute $h = H_2(m, U, pk_A, x_B pk_A) \in Z_q^*$.
(3) Accept the signature and return 1 if and only if the following equation holds:

$$T \stackrel{?}{=} e\left(U + hQ_A, d_B\right). \tag{6}$$

(vi) *CLSDVS-Simulation*. The designated verifier B cannot prove to a third party that a signature $\sigma = (U, T)$ on a message m has been produced by signer A since he can also create an indistinguishable signature σ on m by the following means.

(1) Pick randomly $r' \in Z_q^*$, and compute $U' = r'P$.
(2) Set $h' = H_2(m, U', pk_A, x_B pk_A)$.
(3) Compute $T' = e(U' + h'Q_A, d_B)$.

The signature on the message m is $\sigma' = (U', T')$.

Attack on Yang et al.'s Scheme. Since in CLSDVS-Verify algorithm,

$$T = e(U + hQ_A, d_B) = e(U, d_B)\, e(Q_A, d_B)^h. \quad (7)$$

When one third party W gets $(V_1, V_2) = (x_A pk_B, e(Q_A, d_B))$, W picks a random value $r^* \in Z_q^*$, and computes

$$U^* = r^*P, \qquad h^* = H_2(m^*, U^*, pk_A, V_1), \qquad T^* = V_2^{h^*} e(P_0, r^*Q_B). \quad (8)$$

W can obtain a simulated signature $\sigma^* = (U^*, T^*)$. Because

$$h^* = H_2(m^*, U^*, pk_A, V_1) = H_2(m^*, U^*, pk_A, x_B pk_A),$$
$$T^* = V_2^{h^*} e(P_0, r^*Q_B) = e(h^*Q_A, d_B)\, e(r^*P, d_B) = e(h^*Q_A + U^*, d_B). \quad (9)$$

So, $\sigma^* = (U^*, T^*)$ is a valid signature. Therefore, Lee et al.'s scheme is delegatable.

3.3. Lee et al.'s Scheme. Lee et al.'s strong designated verifier signature scheme [15] is as follows.

Let p and q be two large primes such that $q \mid p - 1$ and g an element of Z_p^* of order q. Let the signer Alice's public key be $y_A = g^{x_A} \bmod p$, where $x_A \in Z_q^*$ is her secret key, and designated verifier Bob's public key $y_B = g^{x_B} \bmod p$, where $x_B \in Z_q^*$ is his secret key. One-way hash function H outputs values in Z_q. Suppose that Alice wants to send a strong designated verifier signature with a message m to Bob.

(i) *Signature Generation.* Alice selects a random value $k \in Z_q^*$. She computes r, s, and t as follows:

$$r = g^k \bmod p, \qquad s = k + x_A r \bmod q, \qquad t = H(m, y_B^s \bmod p). \quad (10)$$

Then, the signature is $\sigma = (r, t)$.

(ii) *Signature Verification.* Upon receiving m and σ, Bob can verify the validity of the signature by checking whether $t = H(m, (r y_A^{r^*})^{x_B} \bmod p)$.

(iii) *Signature Simulation.* Bob can simulate the transcript $\sigma' = (r', t')$ for the message m by selecting a random number $k' \in Z_q^*$ and compute r' and t' as follows:

$$r' = g^{k'} \bmod p, \qquad t' = H\left(m, \left(r' y_A^{r'}\right)^{x_B} \bmod p\right). \quad (11)$$

Attack on Lee et al.'s Scheme. Assume that the signer discloses $y_B^{x_A}$ or the designated verifier $y_A^{x_B} (= y_B^{x_A})$ to any third party T. Given any message m^*, T selects a random number $k^* \in Z_q^*$ and computes

$$r^* = g^{k^*}, \qquad t^* = H\left(m^*, y_B^{k^*} (y_A^{x_B})^{r^*} \bmod p\right). \quad (12)$$

T generates a simulated signature (t^*, r^*). Bob verifies whether $t^* = H(m^*, (r^* y_A^{r^*})^{x_B} \bmod p)$. The verification accepts since

$$H\left(m^*, \left(r^* y_A^{r^*}\right)^{x_B} \bmod p\right) = H\left(m^*, \left(g^{k^*} y_A^{r^*}\right)^{x_B} \bmod p\right) = H\left(m^*, y_B^{k^*} (y_A^{x_B})^{r^*} \bmod p\right). \quad (13)$$

Therefore, Lee et al.'s scheme is delegatable.

3.4. Ki et al.'s Scheme. Ki et al.'s strong designated verifier signature scheme [17] is as follows.

(i) *Setup.* Let G be an additive group and G_T a multiplicative group. Let $e : G \times G \to G_T$ be a symmetric bilinear map, where G and G_T have prime order q. P is a random generator of G. The algorithm selects $s \in Z_q^*$ at random and computes $P_{\text{pub}} \leftarrow sP \in G$. It also selects two collision-resistant cryptographic hash functions, $H_0 : \{0,1\} \to G$ and $H : \{0,1\}^* \to \{0,1\}^\lambda$. The algorithm outputs the master secret key, msk $= s$, and its corresponding public parameters, params $= (G, G_T, q, e, P, P_{\text{pub}}, H_0, H)$.

(ii) *Key-Extract.* For given identity ID, it computes $Q_{\text{ID}} = H_0(\text{ID}) \in G$ and $sk_{\text{ID}} \leftarrow sQ_{\text{ID}}$.

(iii) *IDSig.* For given message $m \in \{0,1\}^*$, verifier's identity ID_V, and signer's secret key $sk_{\text{ID}_S} = sH_0(\text{ID}_S)$, it computes $Q_{\text{ID}_V} \leftarrow H_0(\text{ID}_V) \in G$ and $TK \leftarrow e(sk_{\text{ID}_S}, Q_{\text{ID}_V}) \in G_T$. It selects $r \in Z_q^*$ and computes $\theta \leftarrow rP \in G$ and $k_d \leftarrow e(rP_{\text{pub}}, Q_{\text{ID}_V})$. It computes $\eta \leftarrow H(k_d \parallel TK)$ and $\tau \leftarrow H(\eta \| \theta \| m)$. The signature on a message is $\sigma = (\theta, \tau)$.

(iv) *IDVerify.* For a given signature $\sigma = (\theta, \tau)$, message m, and verifier's secret key sk_{ID_V}, it computes $Q_{ID_S} \leftarrow H_0(ID_S)$, $TK' \leftarrow e(Q_{\text{ID}_S}, sk_{\text{ID}_V})$, $k'_D \leftarrow e(\theta, sk_{\text{ID}_V})$, and $\eta' \leftarrow H(k'_D \parallel TK')$. It tests if $H(\eta' \| \theta \| m) \stackrel{?}{=} \tau$ holds. If the equality holds, then it outputs valid; otherwise, it outputs invalid.

Attack on Ki et al.'s Scheme. Obviously any third party T can generate valid signature when they get $TK = e(sk_{\text{ID}_S}, Q_{\text{ID}_V})$. So, Ki et al.'s scheme is delegatable.

4. Conclusion

Strong designated verifier signatures provide authentication of a message, without, however, having the nonrepudiation property of traditional signatures. They convince one and only one specified recipient that they are valid, but unlike standard digital signature, nobody else can be convinced about their validity or invalidity. The reason is that the designated verifier in these schemes is able to create a signature intended to himself, that is, indistinguishable from a "real" signature. Strong designated verifier signatures fit into various cryptographic applications where privacy preservation is needed. Recently, four strong designated verifier signature schemes are proposed. However, in this work, we show that the four schemes are delegatable. That is to say, in their scheme the signer or the designated verifier can delegate the signing right to any third party by releasing a piece of information related to but different from their secret keys. This enables a third party to simulate the signer's signatures. So, these schemes do not satisfy nondelegatability secure requirement of strong designated verifier signature scheme.

Conflict of Interests

The authors of the paper do not have any conflict of interests.

References

[1] S. Saeednia, S. Kramer, and O. Markovitch, "An efficient strong designated verifier signature schemem," in *Information Security and Cryptology—ICISC 2003*, pp. 40–54, Springer, Berlin, Germany, 2003.

[2] M. Jakobsson, K. Sako, and R. Impagliazzo, "Designated verifier proofs and their applications," in *Advances in Cryptology—EUROCRYPT '96*, vol. 1070 of *Lecture Notes in Computer Science*, pp. 143–154, Springer, 1996.

[3] Y. Lu and G. Tsudik, "Privacy-preserving cloud database querying," *Journal of Internet Services and Information Security*, vol. 1, pp. 5–24, 2011.

[4] N. Gal-oz, T. Grinshpoun, and E. Gudes, "Privacy issues with sharing and computing reputation across communities," *Journal of Wireless Mobile Networks*, vol. 1, pp. 16–34, 2011.

[5] L. Ogiela and M. R. Ogiela, "Fundamentals of cognitive informatica," in *Advances in Cognitive Information Systems*, vol. 17 of *Cognitive System Monographs*, pp. 19–49, Springer, 2012.

[6] X. Huang, W. Susilo, Y. Mu, and F. Zhang, "Short designated verifier signature scheme and its identity-based variant," *International Journal of Network Security*, vol. 6, no. 1, pp. 82–93, 2003.

[7] K. Kumar, G. Shailaja, and A. Saxena, "Identity based strong designated verifier signature scheme," *Informatica*, vol. 18, no. 2, pp. 239–252, 2007.

[8] W. Susilo, F. Zhang, and Y. Mu, "Identity-based strong designated verifier signature schemes," in *Information Security and Privacy*, vol. 3108 of *Lecture Notes in Computer Science*, pp. 313–324, 2004.

[9] J. Zhang and J. Mao, "A novel ID-based designated verifier signature scheme," *Information Sciences*, vol. 178, no. 3, pp. 766–773, 2008.

[10] S. Lal and V. Verma, "Identity Base Strong Designated Verifier Proxy Signature Schemes," Cryptography eprint Archive Report 2006/394, http://eprint.iacr.org/2006/394.

[11] B. Kang, C. Boyd, and E. Dawson, "A novel identity-based strong designated verifier signature scheme," *The Journal of Systems and Software*, vol. 82, no. 2, pp. 270–273, 2009.

[12] B. Kang, C. Boyd, and E. Dawson, "Identity-based strong designated verifier signature schemes: attacks and new construction," *Computers and Electrical Engineering*, vol. 35, no. 1, pp. 49–53, 2009.

[13] J.-S. Lee, J. H. Chang, and D. H. Lee, "Forgery attacks on Kang et al.'s identity-based strong designated verifier signature scheme and its improvement with security proof," *Computers and Electrical Engineering*, vol. 36, no. 5, pp. 948–954, 2010.

[14] Q. Huang, G. Yang, D. S. Wong, and W. Susilo, "Identity-based strong designated verifier signature revisited," *The Journal of Systems and Software*, vol. 84, no. 1, pp. 120–129, 2011.

[15] J.-S. Lee and J. H. Chang, "Comment on Saeednia et al.'s strong designated verifier signature scheme," *Computer Standards & Interfaces*, vol. 31, no. 1, pp. 258–260, 2009.

[16] J. Lee and J. Chang, "Strong designated verifier signature scheme with message recovery," *Advanced Communication Technology*, vol. 1, pp. 801–803, 2007.

[17] J. Ki, J. Y. Hwang, D. Nyang, B.-H. Chang, D. H. Lee, and J.-I. Lim, "Constructing strong identity-based designated verifier signatures with self-unverifiability," *ETRI Journal*, vol. 34, no. 2, pp. 235–244, 2012.

[18] B. Yang, Z. Hu, and Z. Xiao, "Efficient certificateless strong designated verifier signature scheme," in *Proceedings of the International Conference on Computational Intelligence and Security (CIS '09)*, pp. 432–436, IEEE Computer Society, December 2009.