INFORMATICA, 2010, Vol. 21, No. 1, 117–122



# Delegatability of an Identity Based Strong Designated Verifier Signature Scheme *

Xun SUN[1,2], Jianhua LI[2,3], Hu YIN[1], Gongliang CHEN[3]
[1]*Sybase China*
*Shanghai 201203, China*
[2]*Department of Electronic Engineering, Shanghai Jiao Tong University*
*Shanghai 200240, China*
[3]*School of Information Security Engineering, Shanghai Jiao Tong University*
*Shanghai 200240, China*
*e-mail: xun.sun.cn@gmail.com*

Received: June 2008; accepted: October 2008

**Abstract.** In 2007, Kancharla *et al.* proposed an identity-based strong designated verifier signature (IBSDVS) scheme based on bilinear pairings, and claimed it unforgeable and non-delegatable. However, in this paper, we show that this scheme is actually delegatable.

**Keywords:** designated verifier signature scheme, non-delegatability.

## 1. Introduction

In 1996, Jakobsson *et al.* introduced the notion of designated verifier signature (DVS) scheme. In a DVS scheme, the signature provides authentication of a message without providing the non-repudiation property of traditional signatures. From any third party's point of view, a DVS can be generated by both signer (Alice) and the designated verifier (Bob), only Bob can check if the signature is indeed generated by Alice, due to the fact that Bob can always construct a signature designated to himself that is indistinguishable from an original signature by Alice. Consequently, Bob cannot convince other entities about the validity or invalidity of the signatures.

This kind of signature schemes find applications in areas such as call for tenders, electronic voting and distributed contract signing. In a *strong* DVS (Saeednia *et al.*, 2003; Laguillaumie and Vergnaud, 2005) scheme, signature verification requires Bob's secret key to operate correctly.

Lipmaa *et al.* (2004) studied the non-delegatability of DVS schemes. They showed that in most of the previously proposed DVS schemes, Alice can delegate her signing ability with respect to Bob to Charlie by giving some partial information of her secret key to Charlie. This observation is undesirable in practice, if the DVS scheme is used to

*This work is supported by the National Natural Science Foundation of China (60772098 and 60672068) and a national project (C1420061353).

authenticate for a subscriber-only service, delegatability means that Alice can lend her account to Charlie for a moderate amount of money. To capture this issue, Lipmaa *et al.* (2004) proposed and formalized the security notion of non-delegatability.

In 1996, Kancharla *et al.* proposed an identity-based strong DVS (IBSDVS) scheme (hereafter refereed to as the KGS scheme) and proved it existentially unforgeable (against adaptive chosen identity and message attack) based on the bilinear Diffie–Hellman assumption in the random oracle. They also claimed this scheme non-delegatable based on the fact that the signer's secret key is required explicitly in the signing phase.

In this paper, however, we show that the KGS scheme is actually delegatable. Both the signer and the designated verifier can delegate their respective signing and simulation ability to a third party by sending some partial information of their secret keys. The delegatability holds if the Computational Diffie–Hellman (CDH) problem is hard in the underlying group, which we assume naturally in almost all of pairing-based cryptographic schemes.

## 2. Review of the KGS Scheme

In this section, we briefly review the IBSDVS scheme based on bilinear pairings due to Kancharla *et al.* (2007). Formal model of IBSDVS scheme can be found in Susilo *et al.* (2004), Kancharla *et al.* (2007). Its security consists of three aspects: *unforgeability*, *non-transferability* and *non-delegatability*. Definition and properties of bilinear pairings can be found in Boneh and Franklin (2001); Dutta *et al.* (2004). The KGS scheme works as follows.

- `Setup`. PKG first generates two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ both of prime order $q$, for which there is a bilinear pairing $e$: $\mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$, then selects a random generator $P$ of $\mathbb{G}_1$. PKG also selects two cryptographic hash functions $H_1$: $\{0,1\}^* \to \mathbb{G}_1$ and $H_2$: $\{0,1\}^* \times \mathbb{G}_2 \to \mathbb{G}_1$, a random number $s \in \mathbb{Z}_q^*$ and computes $P_{pub} = sP$. $\mathbb{G}_1, \mathbb{G}_2, q, P, P_{pub}, H_1, H_2, e$ are made public and $s$ is kept secret as the master secret key.
- `KeyGen`. On input identity $ID$ and master secret key $s$, compute secret key $S_{ID} = sH_1(ID)$.
- `DeSign`. On input signer $ID_S$'s secret key $S_{ID_S}$, verifier's identity $ID_V$ and message $M$, signer selects $r_1, r_2, r_3 \in_R \mathbb{Z}_q^*$ and computes

$$
\begin{aligned}
&U_1 = r_1 H(ID_V),\\
&U_2 = r_2 H(ID_S),\\
&U_3 = r_1 r_3 H(ID_V),\\
&V = r_3 H + r_1^{-1} S_{ID_S}, \quad \text{where } H = H_2\big(M, e(r_2 H(ID_V), S_{ID_S})\big).
\end{aligned}
$$

  The signature is $\sigma = (U_1, U_2, U_3, V)$.
- `DeVerify`. On input verifier $ID_V$'s secret key $S_{ID_V}$, signer's identity $ID_S$, message $M$ and signature $\sigma = (U_1, U_2, U_3, V)$, verifier first computes $H =$

$H_2(M, e(U_2, S_{ID_V}))$, and evaluates

$$e(U_1, V) \stackrel{?}{=} e(U_3, H)e\big(S_{ID_V}, H(ID_S)\big).$$

$\sigma$ is valid if the equation holds.

- `Simulation`. On input verifier $ID_V$'s secret key $S_{ID_V}$, signer's identity $ID_S$ and message $M$, verifier simulates a signature as follows. He selects $r'_1, r'_2, r'_3 \in_R \mathbb{Z}_q^*$, then computes

$$\begin{aligned}
&U'_1 = r'_1 H(ID_S),\\
&U'_2 = r'_2 H(ID_V),\\
&U'_3 = r'_1 r'_3 H(ID_S),\\
&V' = r'_3 H' + r'^{-1}_1 S_{ID_V}, \quad \text{where } H' = H_2\big(M, e(U'_2, S_{ID_V})\big).
\end{aligned}$$

The simulated signature is then $\sigma' = (U'_1, U'_2, U'_3, V')$.

Correctness of this scheme is easily verified. The authors proved it unforgeable in the random oracle model based on the BDH problem. They also claimed it non-delegatable due to the fact that $S_{ID_S}$ is explicitly involved in the signing algorithm. In the next section, however, we show that this claim is not true.

## 3. Delegatability of the KGS Scheme

Suppose signer has given the tuple $(T_1, T_2) = (rH(ID_V), r^{-1}S_{ID_S})$ to a third party Charlie, where $r$ is chosen randomly from $\mathbb{Z}_q^*$, then Charlie can sign any message $M$ to the verifier $ID_V$ on behalf of the signer as follows. He picks $r_1, r_2, r_3 \in_R \mathbb{Z}_q^*$, and computes

$$\begin{aligned}
&U_1 = r_1 T_1,\\
&U_2 = r_2 H(ID_S),\\
&U_3 = r_1 r_3 T_1,\\
&V = r_3 H + r_1^{-1} T_2, \quad \text{where } H = H_2\big(M, e(r_2 T_1, T_2)\big),
\end{aligned}$$

and sets the signature $\sigma = (U_1, U_2, U_3, V)$.

This signature is valid as verified by Bob, because

$$e(r_2 T_1, T_2) = e(rH(ID_V), r^{-1}S_{ID_S})^{r_2} = e(H(ID_V), S_{ID_S})^{r_2} = e(U_2, S_{ID_V}),$$

and

$$e(U_1, V) = e(r_1 T_1, r_3 H)e(r_1 T_1, r_1^{-1} T_2) = e(U_3, H)e\big(S_{ID_V}, H(ID_S)\big).$$

Furthermore, the signature is of the same distribution as produced by the signer $ID_S$ himself.

Now we have shown that $(T_1, T_2)$ enables Charlie to sign on behalf of signer Alice to the verifier Bob, we still need to show that the tuple $(T_1, T_2)$ does not disclose $S_{ID_S}$ or $S_{ID_V}$ to Charlie – otherwise the delegatability would be trivial. This is also easily seen because, due to the fact that $H$ is modeled as random oracle and $r$ is chosen randomly, computing $S_{ID_S}$ such that $e(T_1, T_2) = e(S_{ID_S}, H(ID_V))$ enables Charlie to solve the CDH problem instance $(H(ID_V), T_1, T_2)$.

Similarly, the verifier can also delegate his simulation ability to Charlie. Therefore the KGS scheme is actually delegatable.

## 4. Conclusion

Recently Kancharla *et al.* proposed an identity-based strong designated verifier signature scheme, and claimed it unforgeable and non-delegatable. In this paper we have shown that their scheme is actually delegatable.

## References

Boneh, D., Franklin, M. (2001). Identity-based encryption from the Weil pairing. In: *Advances in Cryptology – CRYPTO 2001. Lecture Notes in Computer Science*, Vol. 2139. Springer, pp. 213–229.

Dutta, R., Barua, R., Sarkar, P. (2004). Pairing-based cryptographic protocols: A survey. *Cryptology ePrint Archive*.

Jakobsson, M., Sako, K., R. Impagliazzo, R. (1996). Designated verifier proofs and their applications. In: *Eurocrypt 96*, *LNCS*, Vol. 1070. Springer, pp. 143–154.

Kancharla, P.K., Gummadidala, S., Saxena, A. (2007). Identity based strong designated verifier signature scheme. *Informatica*, 18(2), 239–252.

Laguillaumie, F., Vergnaud, D. (2005). Designated verifiers signature: anonymity and efficient construction from any bilinear map. In: *SCN 04*, *LNCS*, Vol. 3352. Springer, pp. 107–121.

Lipmaa, H., Wang, G., Bao, F. (2004). Designated verifier signature schemes: attacks, new security notions and a new construction. In: *ICALP 2005*, *LNCS*, Vol. 3580. Springer, pp. 459–471.

Saeednia, S., Kremer, S., Markowitch, O. (2003). An efficient strong designated verifier signature scheme. In: *ICISC 2003*, *LNCS*, Vol. 2836. Springer, pp. 40–54.

Susilo, W., Zhang, F., Mu, Y. (2004). Identity-based strong designated verifier signature schemes. In: *ACISP 2004*, *LNCS*, Vol. 3108. Springer, pp. 313–324.

**X. Sun** is currently a software engineer in Replication Server Shanghai team at Sybase. His research interests include cryptography and database replication technologies. Xun received his PhD from the Department of Electronic Engineering in Shanghai Jiao Tong University, China.

**J. Li** is now a professor at Department of Electronic Engineering in Shanghai Jiao Tong University. He received the PhD in electronic engineering from Shanghai Jiao Tong University. He is a deputy president of the School of Information Security Engineering in Shanghai Jiao Tong University and an expert of national high technology "863" project. His research interests include information security and computer communication networks.

**H. Yin** is the manager of Replication Server Shanghai team at Sybase. His research interests include database replication technologies. He received his ME in computer science from Shanghai Jiao Tong University.

**G. Chen** is now a professor at School of Information Security Engineering in Shanghai Jiao Tong University. His research interests include number theory and elliptic curve cryptosystems.

# Identifikatoriumi pagrįstos griežtos konstrukcijos parašo patikrinimo teisių perdavimas

Xun SUN, Jianhua LI, Hu YIN, Gongliang CHEN

2007 m. Kancharla ir kt. pasiūlė identifikatoriumi pagrįstą griežtos konstrukcijos parašo patikrinimo algoritmą, naudojantį bitiesinius poravimus ir teigė, kad jis yra nesuklastojamas ir nedeleguojamas. Tačiau šiame straipsnyje parodyta, kad iš tikrųjų šis algoritmas yra deleguojamas.