ISSN 1392 – 124X INFORMATION TECHNOLOGY AND CONTROL, 2011, Vol. 40, No. 4

AN EFFICIENT AND SECURE IDENTITY-BASED STRONG DESIGNATED VERIFIER SIGNATURE SCHEME

Eun-Jun Yoon

School of Computer Engineering, Kyungil University,
33 Buho-Ri, Hayang-Ub, Kyungsan-Si, Kyungsangpuk-Do 712-701, Republic of Korea
e-mail: ejyoon@kiu.ac.kr

crossref http://dx.doi.org/10.5755/j01.itc.40.4.982

Abstract. 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 2008, Zhang and Mao proposed a novel ID- based strong designated verifier signature scheme based on bilinear pairings by combining ID-based cryptosystem with the designated verifier signature. However, Kang et al. pointed out that Zhang-Mao scheme did not satisfy the strong designated verifier signature property and then proposed an efficient ID-based designated verifier signature scheme that is strong and unforgeable. Nevertheless, this paper demonstrates that Kang et al.'s scheme is still vulnerable to universal forgery attacks and then proposes an improved scheme that not only can overcome such forgery attacks but also can provide more efficiency.

Keywords: ID-based cryptography, identity-based signature, bilinear parings, forgery attacks.

1. Introduction

In 1996, Jakobsson et al. [1] first proposed the concept of designated verifier signature schemes. A designated verifier signature scheme is special type of digital signature which provides message authentication without non-repudiation. These signatures have several applications such as in E-voting, call for tenders and software licensing [2–7]. Suppose the signer Alice has sent a designated verifier signature to the designated verifier Bob. In the designated verifier signature scheme, Bob cannot prove to a third party that Alice has created the signature unlike the conventional digital signatures which anyone can verify the validity of a signature using the signer's public key. This is accomplished by the Bob's capability of creating another signature designated to himself which is indistinguishable from Alice's signature. In 2003, Saeednia et al. [8] formalized the notion of strong designated verifier signature scheme based on Jakobsson et al.'s scheme which no third party can verify the validity of a designated verifier signature since the designated verifier's private key is required in the verifying phase.

Up to now, many ID-based strong designated verifier signature schemes based on bilinear pairings by combining ID-based cryptosystem with the designated verifier signature have been proposed [9–18]. In 2004, Laguillaumie et al. [9, 10] constructed two designated verifier signature schemes. The first ID-based strong designated verifier signature scheme was presented by Susilo et al. [11] in 2004. Lipmaa et al. [12] pointed out that Saeednia et al.'s scheme [8] was insecure against delegatability attack. Namely, in Saeednia et al.'s scheme, a signer can delegate his/her signing ability, with respect to a fixed designated verifier, to a third party without disclosing his private key.

Quite recently, Zhang and Mao [13] proposed a novel ID-based strong designated verifier signature scheme. They claimed that their scheme satisfies the property of source hiding. However, Huang et al. [14] showed that Zhang and Mao scheme can lack the source hiding property since the verifier in each of them uses of the signer's public key for doing the verification. Moreover, Kang et al. [15] also pointed out that Zhang-Mao scheme cannot satisfy the strong designated verifier signature property, that is, anyone who intercepts one signature can get some information and verify subsequent signatures. They also proposed an efficient ID-based designated verifier signature scheme that is strong and unforgeable. Unforgeable means that it computationally infeasible to construct a valid ID-based designated verifier signature without the knowledge of the private key of either the signer or the designated verifier. Specially, a universal forgery attack results in the ability to forge signatures for any message.

This paper demonstrates that Kang et al.'s scheme is still vulnerable to universal forgery attacks. An adversary $\mathcal{A}$ can easily generate a valid forged signature (σ^*, U^*) on an arbitrarily chosen message M^* without the secret key of either the signer Alice or the designated verifier Bob in the *Sign* phase. In addition, an improved ID-based strong designated verifier signature scheme is also proposed that not only can overcome such forgery attacks but also can provide more efficiency.

The paper is organized as follows. Section 2 describes background concepts of bilinear pairings, some related mathematical problems, the model for ID-based designated verifier signature scheme, and security properties. Section 3 reviews the Kang et al.'s scheme and then shows its weakness. Section 4 presents the proposed efficient and secure ID-based designated verifier signature and then analyzes its security and efficiency. Finally, Section 5 concludes the paper.

2. Preliminaries

This section introduces the basic concepts of bilinear pairings, some related mathematical problems, the model for ID-based designated verifier signature scheme, and security properties [11–18].

2.1. Bilinear Pairings

Let G_1 be an additive cyclic group with prime order q, G_2 be a multiplicative cyclic group of the same order q. Let $e : G_1 \times G_1 \rightarrow G_2$ be a bilinear mapping with the following properties:

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

A bilinear Diffie-Hellman (BDH) parameter generator is defined as a probabilistic polynomial time algorithm that takes as input a security parameter k and returns a uniformly random tuple (q, G_1, G_2, e, P) of bilinear parameters, including a prime number q of size k, a cyclic additive group G_1 of order q, a multiplicative group G_2 of order q, a bilinear map $e : G_1 \times G_1 \rightarrow G_2$ and a generator P of G_1.

2.2. Computational Problems

Many pairing-based cryptographic schemes are based on the hardness of the following problems. No algorithm is known to be able to solve any of them so far.

1. *Discrete logarithm problem (DLP)*: Given two elements $P, Q \in G_1$, find an integer $a \in Z_q^*$, such that $Q = aP$ whenever such an integer exists.
2. *Computational Diffie-Hellman problem (CDHP)*: For any $a, b \in Z_q^*$, given P, aP, bP, compute abP.
3. *Decisional Diffie-Hellman problem (DDHP)*: For any $a, b, c \in Z_q^*$, given P, aP, bP, cP, decide whether $c = ab \bmod q$.
4. *Bilinear Diffie-Hellman Problem (BDHP)*: Given randomly chosen $P \in G_1$, as well as aP, bP and cP (for unknown randomly chosen $a, b, c \in Z_q$), compute $e(P, P)^{abc}$.
5. *Gap Diffie-Hellman Problem (GDHP)*: A class of problems, where DDHP can be solved in polynomial time but no probabilistic polynomial time algorithm exists which can solve CDHP.
6. *Bilinear Diffie-Hellman (BDH) Assumption*: If $\mathcal{IG}$ is a BDH parameter generator, the advantage $Adv_{\mathcal{IG}}(\mathcal{A})$ that an algorithm $\mathcal{A}$ has in solving the BDH problem is defined to be the probability that the algorithm $\mathcal{A}$ outputs $e(P, P)^{abc}$ on inputs $G_1, G_2, e, P, aP, bP, cP$, where G_1, G_2, e are the output of $\mathcal{IG}$ for sufficiently large security parameter k, P is a random generator of G_1 and a, b, c are random elements of Z_q. The BDH assumption is that $Adv_{\mathcal{IG}}(\mathcal{A})$ is negligible for all efficient algorithms $\mathcal{A}$.

2.3. Model for ID-based Designated Verifier Signature Scheme

In general, an ID-based designated verifier signature scheme consists of five algorithms, namely, Setup, KeyExtract, Sign, Verify and Transcript simulation. These algorithms are defined as follows:

1. *Setup* is a probabilistic polynomial algorithm that takes a security parameter k as input and returns the system parameters $params$ and master key *master-key*.
2. *KeyExtract* is a probabilistic polynomial algorithm that takes *params*, *master-key* and an arbitrary string $ID \in \{0, 1\}^*$ as inputs. It returns a private key S_{ID}. Here ID is the signer's identity and will be used as the signer's public key.
3. *Sign* is a probabilistic polynomial algorithm that takes *params*, the signer's private key S_{ID_A}, a message M and the designated verifier's public key Q_{ID_V} as inputs. The algorithm outputs a signature σ on the message M.
4. *Verify* is a deterministic polynomial algorithm that takes *params*, the signer's identity ID_S, a message M, the designated verifier's identity

Shared Information: $(Q_{ID_A}, Q_{ID_B}, G_1, G_2, P, P_{pub}, H_1, H_2, e, q)$
Information held by Alice: $S_{ID_A} = sH_1(ID_A)$
Information held by Bob: $S_{ID_B} = sH_1(ID_B)$

Alice **Bob**

Sign phase:
Choose M
$r \in Z_q^*$
$U = rQ_{ID_A}$
$\sigma = H_2(M, e(rQ_{ID_B}, S_{ID_A}))$

$$\xrightarrow{M, (U, \sigma)}$$

Verify phase:
Verify $\sigma \overset{?}{=} H_2(M, e(U, S_{ID_B}))$

Figure 1. Kang et al.'s ID-based designated verifier signature scheme

ID_V and private key S_{ID_V}, and the signature σ as inputs, then it outputs either *accept* or *reject* as the verification decision.

5. *Transcript simulation* is a deterministic polynomial algorithm. The designated verifier runs this algorithm to produce identically distributed transcripts which are indistinguishable from the signature produced by the signer.

2.4. Security Properties

The ID-based designated verifier signature scheme should satisfy the following security properties.

1. *Correctness:* If the signer properly produces an ID-based designated verifier signature by the *Sign* algorithm, the produced signature must be accepted by the verifying algorithm.

2. *Strongness*: To verify a signature, the secret key of the designated verifier should be involved in the verification step.

3. *Unforgeability*: It is computationally infeasible to construct a valid ID-based designated verifier signature without the knowledge of the private key of either the signer or the designated verifier. Specially, a universal forgery attack results in the ability to forge signatures for any message.

4. *Non-Transferability:* The non-transferability ensures that any designated verifier can produce an indistinguishable signature from the one generated by the real signer. That is, the designated verifier cannot prove to a third party that the signature was produced by the signer or the designated verifier.

5. *Source hiding:* Given a message M and a valid ID-based designated verifier signature on M, it is infeasible to determine who produced this signature from the original signer or the designated verifier, even if one knows all the secret keys of both the signer and the verifier.

3. Universal Forgery Attacks on Kang et al.'s Scheme

This section reviews Kang et al.'s ID-based designated verifier signature scheme [15] and then shows the scheme is suffer from universal forgery attacks.

3.1. Kang et al.'s ID-based Designated Verifier Signature Scheme

Fig. 1 illustrates Kang et al.'s ID-based designated verifier signature scheme. Their protocol proceeds as follows:

1. *Setup*: In this phase, the PKG (private key generation center) chooses a gap Diffie-Hellman group G_1 of prime order q and a multiplicative group G_2 of the same order and a bilinear map $e : G_1 \times G_1 \rightarrow G_2$, together with an arbitrary generator $P \in G_1$. Then it chooses a random value $s \in Z_q^*$ as the master secret key and computes the corresponding public key $P_{pub} = sP$. $H_1(\cdot)$ and $H_2(\cdot)$ are two cryptographic hash functions, with $H_1 : \{0,1\}^* \rightarrow G_1$ and H_2 :

Opened Information: $(Q_{ID_A}, Q_{ID_B}, G_1, G_2, P, P_{pub}, H_1, H_2, e, q)$
Information held by Bob: $S_{ID_B} = sH_1(ID_B)$

Adversary **Bob**

Sign phase:
Choose fake M^*
$U^* = P$
$\sigma^* = H_2(M^*, e(P_{pub}, Q_{ID_B}))$

$$\xrightarrow{M^*, (U^*, \sigma^*)}$$

Verify phase:
Verify $\sigma^* \overset{?}{=} H_2(M^*, e(U^*, S_{ID_B}))$

Figure 2. Universal forgery attacks on Kang et al.'s scheme

$\{0,1\}^* \times G_2 \rightarrow G_1$. The system public parameters are $(G_1, G_2, P, P_{pub}, H_1, H_2, e, q)$ and the master secret key is s.

2. *KeyExtract*: Given an identity ID, PKG computes $S_{ID} = sH_1(ID)$ and sends it to the user with identity ID. We remark $Q_{ID} = H_1(ID)$ as the public key of the user with identity ID.

3. *Sign*: Given a secret key S_{ID_A} of the signer Alice, the public keys Q_{ID_A}, Q_{ID_B} of the signer Alice and designated verifier Bob, respectively, and the signed message M, the signer Alice chooses a random number $r \in Z_q^*$ and computes

$$U = rQ_{ID_A}$$

$$\sigma = H_2(M, e(rQ_{ID_B}, S_{ID_A})).$$

Alice sends the resulting signature (σ, U) to the designated verifier Bob.

4. *Verify*: On receiving the designated verifier signature (σ, U), the designated verifier Bob accepts the signature if and only if

$$\sigma = H_2(M, e(U, S_{ID_B}))$$

5. *Transcript simulation*: Bob chooses a random number $r' \in Z_q^*$ and computes

$$U' = r'Q_{ID_A}$$

$$\sigma' = H_2(M, e(U', S_{ID_B}))$$

Obviously, (σ', U') satisfies the verification.

3.2. Universal Forgery Attacks

Kang et al.'s scheme is not secure to the universal forgery attacks. An adversary $\mathcal{A}$ can easily generate a signature (σ^*, U^*) on an arbitrarily chosen message M^* without the secret key of either the signer Alice or the designated verifier Bob in the *Sign* phase as follows:

1. $\mathcal{A}$ obtains the system public parameters $(G_1, G_2, P, P_{pub}, Q_{ID_B}, H_1, H_2, e, q)$ from PKG.
2. $\mathcal{A}$ chooses an arbitrarily fake message M^*.
3. $\mathcal{A}$ lets $U^* = P$, where P is generator.
4. $\mathcal{A}$ computes $\sigma^* = H_2(M^*, e(P_{pub}, Q_{ID_B}))$, where $P_{pub} = sP$ is the public key of PKG and Q_{ID_B} is the public key of designated verifier Bob.
5. $\mathcal{A}$ sends the resulting signature (σ^*, U^*) to the designated verifier Bob.

On receiving the designated verifier signature (σ^*, U^*) in the *Verify* phase, the designated verifier Bob will check whether the following verification equation holds

$$\sigma^* = H_2(M^*, e(U^*, S_{ID_B})). \qquad (1)$$

We can see that the designated verifier Bob will accept the forged signature and the fake message M^* because the above verification equation (1) is always satisfied as follows:

$$\begin{aligned} \sigma^* &= H_2(M^*, e(P_{pub}, Q_{ID_B})) \\ &= H_2(M^*, e(sP, Q_{ID_B})) \\ &= H_2(M^*, e(P, sQ_{ID_B})) \\ &= H_2(M^*, e(U^*, S_{ID_B})). \end{aligned}$$

Shared Information: $(Q_{ID_A}, Q_{ID_B}, G_1, G_2, P, P_{pub}, H_1, H_2, e, q)$
Information held by Alice: $S_{ID_A} = sH_1(ID_A)$
Information held by Bob: $S_{ID_B} = sH_1(ID_B)$

Alice **Bob**

Sign phase:
Choose M
$r \in Z_q^*$
$\sigma = H_2(M, e(rQ_{ID_B}, S_{ID_A}))$

$$\xrightarrow{\quad M, (r, \sigma) \quad}$$

Verify phase:
Verify $\sigma \overset{?}{=} H_2(M, e(S_{ID_B}, rQ_{ID_A}))$

Figure 3. The proposed ID-based designated verifier signature scheme

As a result, Kang et al.'s scheme is not secure to the above universal forgery attacks. Fig. 2 illustrates the universal forgery attacks on Kang et al.'s scheme.

4. The Proposed ID-based Designated Verifier Signature Scheme

This section proposes an improvement of Kang et al.'s scheme that can withstand the above universal forgery attack and then analyzes the security and efficiency of the proposed scheme.

4.1. The Proposed Scheme

The proposed ID-based designated verifier signature scheme has five phases: *Setup*, *KeyExtract*, *Sign*, *Verify*, and *Transcript simulation*. The *Setup* and *KeyExtract* phases are the same as those of Kang et al.'s scheme. Therefore, the last three phases are described only briefly. Fig. 3 illustrates the proposed ID-based designated verifier signature scheme.

1. *Sign*: Given a secret key S_{ID_A} of the signer Alice, the public keys Q_{ID_A}, Q_{ID_B} of the signer Alice and designated verifier Bob, respectively, and the signed message M, the signer Alice chooses a random number $r \in Z_q^*$ and computes

$$\sigma = H_2(M, e(rQ_{ID_B}, S_{ID_A})).$$

Alice sends the resulting signature (σ, r) to the designated verifier Bob.

2. *Verify*: On receiving the designated verifier signature (σ, r), the designated verifier Bob accepts the signature if and only if

$$\sigma = H_2(M, e(S_{ID_B}, rQ_{ID_A})) \qquad (2)$$

3. *Transcript simulation*: Bob chooses a random number $r' \in Z_q^*$ and computes

$$\sigma' = H_2(M, e(S_{ID_B}, r'Q_{ID_A}))$$

Obviously, (σ', r') satisfies the verification.

4.2. Security Analysis

This subsection analyzes the security of the proposed ID-based designated verifier signature scheme. The ID-based designated verifier signature scheme should satisfy the following security properties: *Correctness*, *Strongness*, *Unforgeability*, *Source hiding*, *Non-transferability*, and *Universal forgery attack*.

1. *Correctness*: A properly produced ID-based designated verifier signature scheme must be accepted by the signature verification algorithm. The following equation proofs the correctness of the above verification equation (2) in the proposed *Verify* phase:

$$\begin{aligned}\sigma &= H_2(M, e(rQ_{ID_B}, S_{ID_A}))\\ &= H_2(M, e(rQ_{ID_B}, sQ_{ID_A}))\\ &= H_2(M, e(sQ_{ID_B}, rQ_{ID_A}))\\ &= H_2(M, e(S_{ID_B}, rQ_{ID_A})).\end{aligned}$$

2. *Strongness*: To verify a signature, the secret key of the designated verifier should be involved in the verification step. In the proposed scheme, the designated verifier Bob has to use his secret key $S_{ID_B} = sQ_{ID_B}$ during the verification. Nobody can get any useful information to signature verification from intercepted signatures. Therefore,

Table 1. Comparison of computational costs

Scheme	Length	Signing cost	Verifying cost
Susilo-scheme [11]	$2\|G_1\| + \|H\|$	$1C_p + 2C_m + 1C_e + 1C_h + 1C_i$	$2C_p + 1C_m + 2C_e + 1C_h$
Kumar-scheme [17]	$4\|G_1\|$	$1C_p + 5C_m + 1C_h + 1C_i$	$4C_p + 1C_h$
Zhang-scheme [13]	$3\|G_1\|$	$4C_m + 1C_h + 1C_i$	$3C_p + 1C_h$
Kang-scheme [15]	$2\|G_1\|$	$1C_p + 1C_m + 1C_h$	$1C_p + 1C_h$
Proposed-scheme	$1\|G_1\| + \|k\|$	$1C_p + 1C_h$	$1C_p + 1C_h$

C_p: pairing operation, C_m: multiplication in G_1, C_e: exponentiation in G_2
C_h: hash operation, C_i: inverse operation, $|X|$: bit length of X

the proposed scheme is a strong designated verifier scheme.

3. *Unforgeability*: It is computationally infeasible to construct a valid ID-based designated verifier signature without the knowledge of the private key of either the signer or the designated verifier. In the proposed scheme, it is not possible to construct the forged signature $\sigma^* = H_2(M^*, e(S_{ID_B}, r^*Q_{ID_A}))$ against an arbitrarily chosen fake message M^* and a random number r^* without the knowledge of either the signer secret key $S_{ID_A} = sQ_{ID_A}$ or the verifier secret key $S_{ID_B} = sQ_{ID_B}$. Therefore, the proposed signature is unforgeable.

4. *Source hiding*: Given a message M and a valid ID-based designated verifier signature on M, it is infeasible to determine who produced this signature from the original signer or the designated verifier, even if one knows all the secret keys. In the proposed scheme, even if the signer secret key $S_{ID_A} = sQ_{ID_A}$ and the verifier secret key $S_{ID_B} = sQ_{ID_B}$ are given to the third party, he/she cannot identify whether the signer or the designated verifier has produced the signature (σ, r) because the signature is generated as follows:

$$\begin{aligned}\sigma &= H_2(M, e(rQ_{ID_B}, S_{ID_A}))\\ &= H_2(M, e(S_{ID_B}, rQ_{ID_A})).\end{aligned}$$

   Therefore, the proposed scheme provides source hiding.

5. *Non-transferability*: The non-transferability ensures that any designated verifier can produce an indistinguishable signature from the one generated by the real signer. That is, the designated verifier cannot prove to a third party that the signature was produced by the signer or the designated verifier. In the proposed scheme, the non-transferability property can be achieved by the above *source hiding* property and the *transcript simulation* algorithm since the transcripts simulated by the designated verifier Bob are indistinguishable from those that he receives from the real signer Alice. In fact, if (σ, r) is a valid signature, then the probability of its generation by Alice or Bob are identical as $\frac{1}{q-1}$. Therefore, the proposed scheme provides non-transferability.

6. *Universal forgery attack*: A universal forgery attack results in the ability to forge signatures for any message. In Section 3.2, we proved that Kang et al.'s scheme is not secure to the universal forgery attack. In Kang et al.'s scheme, upon receiving (σ^*, U^*) from Alice, the designated verifier Bob directly uses the received U^* without any computation to verify the signature by using the following verification equation:

$$\sigma^* = H_2(M^*, e(U^*, S_{ID_B})).$$

   Therefore, an adversary $\mathcal{A}$ can easily generate a signature (σ^*, U^*) on an arbitrarily chosen message M^* without the secret key of either the signer Alice or the designated verifier Bob. In the proposed scheme, the designated verifier Bob does not use directly the received r unlike Kang et al.'s scheme. That is, upon receiving (r, σ) from Alice, the designated verifier Bob computes rQ_{ID_A} by using r and Q_{ID_A}, where Q_{ID_A} is the signer Alice's identity. It can simply protect the proposed universal forgery attack because the adversary $\mathcal{A}$ cannot compute a valid $e(rQ_{ID_B}, S_{ID_A})$ without knowing the signer's private key S_{ID_A} or the verifier's private key S_{ID_B}. Therefore, the proposed signature is secure to the universal forgery attack.

4.3. Efficiency Analysis

This subsection gives a performance comparison of the proposed scheme with the previous related ID-based designated verifier signature schemes [11, 13, 17] and Kang et al.'s scheme [15] based on the length of the signature and the required computational cost. Table 1 shows the comparison results of the computational costs of the proposed scheme and of various ID-based designated verifier signature schemes. In order to compare the computational workload, we considered the number of pairing operations, multiplications, exponentiations, hash operations, and inverse operations. In the *sign* and *verify* phases, Kang et al.'s scheme requires the smallest computational costs among the previously proposed schemes. By contrast, the proposed scheme requires one pairing and one hash operations in the *sign* phase. The computation costs of the *verify* phase are the same as those in Kang et al.'s scheme. The size of signature is only $1|G_1| + |k|$, where $|G_1|$ is the bit length of element in G_1 and $|k|$ is the bit length of the size of a prime number q. Therefore, as in Table 1, we know that on the whole, the proposed scheme is more efficient compared with the previous schemes from the literature [11, 13, 15, 17].

5. Conclusions

This paper demonstrated that Kang et al.'s ID-based strong designated verifier signature scheme is vulnerable to universal forgery attacks unlike their claims. To overcome such forgery attacks and provide more efficiency, an efficient and secure ID-based strong designated verifier signature scheme is proposed. As a result, the proposed scheme has advantage of low communication and computational cost compared with related schemes.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments in improving our manuscript. This research was supported by the IT Research and Development program of MKE/KEIT (10035245) and the ITRC program of MKE/NIPA (NIPA-2010-C1090-1021-0002).

References

[1] **M. Jakobsson, K. Sako, R. Impagliazzo.** Designated verifier proofs and their applications. *In: Advances in Eurocrypt'96*. LNCS 1070, Springer-Verlag, 1996, 143–154.

[2] **C.C. Lee, Y.F. Chang.** On security of a practical three-party key exchange protocol with round efficiency. *Information Technology and Control*, 2008, 37(4), 333–335.

[3] **C.C. Lee, I.E. Liao, M.S. Hwang.** An extended certificate-based authentication and security protocol for mobile networks. *Information Technology and Control*, 2009, 38(1), 61–66.

[4] **C.C. Lee, T.C. Lin, M.S. Hwang.** A Key Agreement Scheme for Satellite Communications. *Information Technology and Control*, 2010, 39(1), 43–47.

[5] **J.W. Lo, S.C. Lin, M.S. Hwang.** A Parallel Password-Authenticated Key Exchange Protocol for Wireless Environments. *Information Technology and Control*, 2010, 39(2), 146–151.

[6] **E. Sakalauskas, A. Katvickis, G. Dosinas.** Key Agreement Protocol over the Ring of Multivariate Polynomials. *Information Technology and Control*, 2010, 39(1), 51–54.

[7] **J.L. Tsai.** Weaknesses and Improvement of Hsu-Chuang's User Identification Scheme. *Information Technology and Control*, 2010, 39(1), 48–50.

[8] **S. Saeednia, S. Kramer, O. Markovitch.** An efficient strong designated verifier signature scheme. *In: ICISC 2003*, LNCS 2971, Springer-Verlag, 2003, 40-54.

[9] **F. Laguillaumie, D. Vergnaud.** Multi-designated verifier signatures. *In: ICICS 2004*, LNCS 3269, Springer-Verlag, 2004, 495–507.

[10] **F. Laguillaumie, D. Vergnaud.** Designated verifier signatures: anonymity and efficient construction from any bilinear map. *In: SCN2004*, LNCS, 3352, Springer-Verlag, 2004, 105–119.

[11] **W. Susilo, F. Zhang, Y. Mu.** Identity-based strong designated verifier signature schemes. *In: ACISP 2004*. LNCS 3108, Springer-Verlag, 2004, 313–324.

[12] **H. Lipmaa, G. Wang, F. Bao.** Designated verifier signature schemes: attacks, new security notions and new construction. *In: ICALP 2005*, LNCS, 3580, Springer-Verlag, 2005, 459–471.

[13] **J. Zhang, J. Mao.** A novel ID-based designated verifier signature scheme. *Information Sciences*, 2008, 178, 766–773.

[14] **X. Huang, W. Susilo, Y. Mu, F. Zhang.** Short designated verifier signature scheme and its identity-based variant. *International Journal of Network Security*, 2008, 6(1), 82–93.

[15] **B. Kang, C. Boyd, E. Dawson.** Identity-based strong designated verifier signature schemes: attacks and new construction. *Computers & Electrical Engineering*, 2009, 35(1), 49–53.

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

[17] **K. Kumar, G. Shailaja, A. Saxena.** Identity based strong designated verifier signature scheme. *Cryptography eprint Archive Report 2006/134*. Available at http://eprint.iacr.org/complete/2006/134.pdf.

[18] **L. Sunder, V. Vandani.** Identity base strong designated verifier proxy signature schemes. *Cryptography eprint Archive Report 2006/394*. Available at http://eprint.iacr.org/complete/2006/394.pdf.

Received September 2010.