ISSN 1392–124X (print), ISSN 2335–884X (online) INFORMATION TECHNOLOGY AND CONTROL, 2013, Vol. 42, No. 3

Secure Universal Designated Verifier Signature and its Variant for Privacy Protection

Han-Yu Lin

Department of Computer Science and Engineering,
National Taiwan Ocean University,
Keelung, Taiwan
e-mail: lin.hanyu@msa.hinet.net

crossref http://dx.doi.org/10.5755/j01.itc.42.3.3532

Abstract. Based on the bilinear inverse Diffie-Hellman problem (BIDHP), we first propose a provably secure probabilistic signature scheme. Furthermore, we extend it into two universal designated verifier signature (UDVS) schemes under the same computational assumption. The first one is a conventional UDVS scheme for one designated verifier while the other is designed for cooperative multi-verifier. UDVS schemes aim at protecting the privacy of signature holders and have practical benefits to the applications, e.g., the certificate for medical records and income summary, etc. The comparison results demonstrate that the signature generation and designation of our scheme are both pairing-free, which could benefit the application of devices with constrained computation. We also give formal security proofs of unforgeability against existential forgery under adaptive chosen-message attacks (EF-CMA) in the random oracle model.

Keywords: probabilistic signature; universal designated verifier signature; bilinear inverse Diffie-Hellman problem; privacy-preserving; random oracle.

1. Introduction

In the digitalized world, digital signature schemes [1-5] are commonly applied mechanisms, which can be used to ensure the integrity, the authenticity [6] and the non-repudiation [7]. To be precise, any third party can first acquire the signer's public key and then verify the corresponding signatures. Since only the one who owns the private key can create a valid signature, the signer cannot repudiate his generated signatures later. There are usually two types of digital signature schemes, i.e., deterministic [1] and probabilistic [6]. A deterministic signature scheme has the unique signature for the same message while a probabilistic one always generates different signatures for an identical message.

Consider the privacy issue in some special applications such as the electronic voting [8, 9]. It is not desired for anyone to verify the resulting signature. To realize the notion, in 1990, Chaum and Antwerpen [10] proposed the undeniable signature scheme in which a verifier must interactively cooperate with the signer to verify generated signatures. In other words, the signer has the ability to determine which verifier is able to check the validity of his signatures. Hence the privacy requirement is fulfilled in the undeniable signature scheme.

In 1996, Jakobsson *et al*. [11] introduced the idea of non-interactive designated verifier proof and addressed the designated verifier signature (DVS) scheme without non-repudiation. In a DVS scheme, the signer does not have to participate in every signature verification process. When creating the DVS, the signer directly incorporates the designated verifier's public key with the signing process. A special property of the DVS scheme is that the designated verifier can use his private key to generate another valid DVS intended for himself, which is referred to as the transcript simulation. Owing to this property, the designated verifier cannot persuade any third party of the DVS's authenticity. Consequently, only the intended verifier will believe the signature's validity. Note that although the DVS is publicly verifiable with the designated verifier's public key, it is difficult for any third party to identify the actual signer and verify the authenticity for a given DVS.

In 2003, however, Wang [12] and Saeednia *et al*. [13] separately found out the security weakness in Jakobsson *et al*.'s scheme. Saeednia *et al*. not only gave a countermeasure, but also further proposed the strong designated verifier signature (SDVS) scheme which removes the property of public verifiability by taking the designated verifier's private key as a crucial

parameter in the signature verification equality. Therefore, without the designated verifier's private key, no one can even check the validity of an SDVS. Since then, lots of researchers have devoted themselves to the study of SDVS variants [14-19].

The universal designated verifier signature (UDVS) schemes were introduced by Steinfeld *et al*. [20, 21], in which the signer and the signature holder are different persons. Focusing on protecting the signature holders' privacy, a UDVS scheme allows any signature holder to non-interactively designate a publicly verifiable signature (PV-signature) to an intended verifier using the verifier's public key. The designated verifier can validate the UDVS with his private key, but can not transfer the conviction to any third party, which is also referred to as non-transferability. The UDVS schemes are useful in stopping the verifier from arbitrarily disseminating the signature and therefore applicable to the applications such as the certificate for medical records and income summary. In 2005, Zhang *et al*. [22] proposed a UDVS scheme based on the strong Diffie-Hellman problem (SDHP). In 2008, Huang *et al*. [23] proposed another UDVS scheme based on the gap bilinear Diffie-Hellman problem (GBDHP). Both Zhang *et al*.'s and Huang *et al*.'s schemes provide the provable security. Nevertheless, the computational costs of their schemes are rather high and many similar variants [24-26] only consider the conventional setting of single designated verifier.

1.1. Our contributions

It is known that a UDVS scheme can be extended from any general digital signature scheme, meaning that a security weakness of a digital signature will also be inherited by its UDVS extension. It is therefore crucial to construct a UDVS scheme from secure probabilistic signatures which provide randomness with respect to even identical messages. In this paper, we first propose a novel and secure probabilistic signature based on the bilinear inverse Diffie-Hellman problem (BIDHP) and then extend it into two UDVS schemes. One is a conventional UDVS scheme allowing one intended verifier to validate the signature while the other is a universal designated multi-verifier signature (UDMVS) scheme in which all verifiers must cooperatively check the signature. Compared with previous schemes, the proposed schemes are especially suitable for computation-constrained devices, since the signature generation and designation are pairing-free. Moreover, the formal security proofs of unforgeability against existential forgery under adaptive chosen-message attacks (EF-CMA) are proved in the random oracle model.

2. Preliminaries

In this section, we briefly review some used security notions and the computational assumptions.

Bilinear Pairing

Let $(\boldsymbol{G}_1, +)$ and $(\boldsymbol{G}_2, \times)$ be two groups of the same prime order q and e: $\boldsymbol{G}_1 \times \boldsymbol{G}_1 \rightarrow \boldsymbol{G}_2$ a bilinear map which satisfies the following properties:

i. Bilinearity:

$e(P_1 + P_2, Q) = e(P_1, Q)\, e(P_2, Q)$;

$e(P, Q_1 + Q_2) = e(P, Q_1)\, e(P, Q_2)$;

ii. Non-degeneracy:

If P is a generator of $\boldsymbol{G}_1$, then $e(P, P)$ is a generator of $\boldsymbol{G}_2$.

iii. Computability:

Given $P, Q \in \boldsymbol{G}_1$, the value of $e(P, Q)$ can be efficiently computed by a polynomial-time algorithm.

Bilinear Diffie-Hellman Problem; BDHP

The BDHP is, given $P, aP, bP, cP \in \boldsymbol{G}_1$ for some $a, b, c \in Z_q$, to compute $e(P, P)^{abc} \in \boldsymbol{G}_2$.

Bilinear Diffie-Hellman (BDH) Assumption

For every probabilistic polynomial-time algorithm $\mathcal{A}$, every positive polynomial $F(\cdot)$ and all sufficiently large k, the algorithm $\mathcal{A}$ can solve the BDHP with an advantage of at most $1/F(k)$, i.e.,

$$\Pr[\mathcal{A}(P, aP, bP, cP) = e(P, P)^{abc}; a, b, c \leftarrow Z_q, (P, aP, bP, cP) \leftarrow \boldsymbol{G}_1{}^4] \le 1/F(k).$$

The probability is taken over the uniformly and independently chosen instance and over the random choices of $\mathcal{A}$.

Definition 1. *The (t, ε)-BDH assumption holds if there is no polynomial-time adversary that can solve the BDHP in time at most t and with an advantage ε.*

Bilinear Inverse Diffie-Hellman Problem; BIDHP

The BIDHP is, given $P, aP, bP \in \boldsymbol{G}_1$ for some $a, b \in Z_q$, to compute $e(P, P)^{a^{-1}b} \in \boldsymbol{G}_2$.

Bilinear Inverse Diffie-Hellman (BIDH) Assumption

For every probabilistic polynomial-time algorithm $\mathcal{A}$, every positive polynomial $F(\cdot)$ and all sufficiently large k, the algorithm $\mathcal{A}$ can solve the BIDHP with an advantage of at most $1/F(k)$, i.e.,

$$\Pr[\mathcal{A}(P, aP, bP) = e(P, P)^{a^{-1}b}; a, b \leftarrow Z_q, (P, aP, bP) \leftarrow \boldsymbol{G}_1{}^3] \le 1/F(k).$$

The probability is taken over the uniformly and independently chosen instance and over the random choices of $\mathcal{A}$.

Definition 2. *The (t, ε)-BIDH assumption holds if there is no polynomial-time adversary that can solve the BIDHP in time at most t and with an advantage ε.*

In fact, BIDHP is one variation of BDHP and both are polynomial-time equivalent. Interested readers could refer to the proof in Zhang *et al.*'s literature [27].

3. Probabilistic Signature Scheme based on BIDHP

In this section, we first address involved parties and composed algorithms of our proposed probabilistic signature scheme and then give a detailed construction.

3.1. Involved parties

A probabilistic signature scheme has two involved parties: a signer and a verifier. Each one is a probabilistic polynomial-time Turing machine (PPTM). The signer generates a publicly verifiable signature (PV-signature) such that the verifier can validate it with signer's public key.

3.2. Algorithms

The proposed scheme consists of three algorithms (including Setup, PSG and PSV). We describe these algorithms as follows:

- **Setup:** Taking as input 1^k where k is a security parameter, the algorithm generates the system's public parameters *params*.
- **PV-Signature-Generation (PSG):** The PSG algorithm takes as input the system parameters *params*, a message and the private key of signer. It generates a PV-signature Ω.
- **PV-Signature-Verification (PSV):** The PSV algorithm takes as input the system parameters *params*, a PV-signature Ω along with the corresponding message m, and the public key of signer. It outputs **True** if Ω is a valid PV-signature for m. Otherwise, the error symbol ¶ is returned as a result.

3.3. Construction of Probabilistic Signature Scheme

We detail the construction of our probabilistic signature scheme as follows:

- **Setup:** Taking as input 1^k, the system authority (SA) selects two groups $(\boldsymbol{G}_1, +)$ and $(\boldsymbol{G}_2, \times)$ of the same prime order q where $|q| = k$. Let P be a generator of order q over $\boldsymbol{G}_1$, e: $\boldsymbol{G}_1 \times \boldsymbol{G}_1 \rightarrow \boldsymbol{G}_2$ a bilinear pairing and h_1: $\boldsymbol{G}_1 \rightarrow \boldsymbol{G}_1$ and h_2: $\{0, 1\}^* \times \boldsymbol{G}_1 \rightarrow Z_q$ collision resistant hash functions. The system publishes the public parameters *params* = $\{\boldsymbol{G}_1, \boldsymbol{G}_2, q, P, e, h_1, h_2\}$. Each user U_i chooses his private key $x_i \in Z_q$ and registers the public key as $Y_i = x_i P$.
- **PV-Signature-Generation (PSG):** Let U_s be the signer. For signing a message $m \in_R \{0, 1\}^*$, U_s chooses $r \in_R Z_q$ to compute

$$R = rP, \tag{1}$$

$$T = x_s^{-2} h_1(R), \tag{2}$$

$$\rho = r + h_2(m, R)\, x_s^{-1} \bmod q. \tag{3}$$

The PV-signature for the message m is $\Omega = (R, T, \rho)$.

- **PV-Signature-Verification (PSV):** To check the validity of the PV-signature $\Omega = (R, T, \rho)$, anyone can verify whether

$$e(\rho P - R, h_1(R)) = e(Y_s, h_2(m, R)T). \tag{4}$$

If the quality holds, the PV-signature is valid. We show that the verification of Eq. (4) works correctly. From the left-hand side of Eq. (4), we have

$$\begin{aligned} & e(\rho P - R, h_1(R)) \\ &= e(rP + h_2(m, R)\, x_s^{-1} P - R, h_1(R)) \\ &= e(h_2(m, R)\, x_s^{-1} P, x_s^2 T) \qquad \text{(by Eq. (2))} \\ &= e(Y_s, T)^{h_2(m, R)} \\ &= e(Y_s, h_2(m, R)T) \end{aligned}$$

which leads to the right-hand side of Eq. (4).

4. Extensions into UDVS and UDMVS schemes

In this section, we present UDVS and UDMVS schemes based on the proposed probabilistic signature scheme. We first address involved parties and composed algorithms of our UDVS/UDMVS scheme and then give concrete constructions.

4.1. Involved parties

A conventional UDVS scheme has three involved parties including a signer, a designator (signature holder) and a designated verifier. Unlike the conventional UDVS scheme, UDMVS scheme has designated multi-verifier, say, consisting of n verifiers. In our proposed UDVS and UDMVS schemes, each party is a probabilistic polynomial-time Turing machine (PPTM). The signer will generate a PV-signature and send it along with the message to the designator. After validating the PV-signature, the designator further creates a designated verifier/multi-verifier signature (DV/DMV-signature) and deliveries it together with the message to the designated verifier/multi-verifier. Consequently, the DV/DMV-signature can only be verified by the designated verifier/multi-verifier with his/their private key(s). Besides, the designated verifier/multi-verifier can not transfer the conviction to any third party, since he/they is/are also capable of generating another computationally indistinguishable transcript.

4.2. Algorithms

The proposed UDVS/UDMVS scheme consists of five algorithms (including Setup, PSG, PSV, DSG and

DSV). The first three algorithms are defined the same as those in our probabilistic signature scheme. We only describe others as follows:

- **DV-Signature-Generation (DSG):** The DSG algorithm takes as input a PV-signature Ω along with the corresponding message m, and the public key of designated verifier. It generates a DV-signature δ.
- **DV-Signature-Verification (DSV):** The DSV algorithm takes as input a DV-signature δ along with the corresponding message m, the private key of the designated verifier, and the public key of signer. It outputs **True** if δ is a valid DV-signature for m. Otherwise, the error symbol ¶ is returned as a result.
- **DMV-Signature-Generation (DMSG):** The DMSG algorithm takes as input a PV-signature Ω along with the corresponding message m, and the public keys of designated multi-verifier. It generates a DMV-signature δ.
- **DMV-Signature-Verification (DMSV):** The DMSV algorithm takes as input a DMV-signature δ along with the corresponding message m, the private keys of the designated multi-verifier, and the public key of signer. It outputs **True** if δ is a valid DMV-signature for m. Otherwise, the symbol is returned as a result.

4.3. Concrete construction of UDVS scheme

We demonstrate the proposed UDVS scheme in the subsection. This scheme is a conventional UDVS which only allows the signature holder to solely designate the PV-signature to one intended designated verifier without further interactions. For simplicity, we only detail phases of DSG and DSV below:

- **DV-Signature-Generation (DSG):** Let U_v be the designated verifier with public key Y_v. To create a DV-signature for a given message m and its PV-signature $\Omega = (R, T, \rho)$, the designator computes

  $$W = \rho Y_v, \quad (5)$$

  and then deliveries the DV-signature $\delta = (R, T, W)$ along with the corresponding message m to U_v.
- **DV-Signature-Verification (DSV):** Upon receiving (δ, m), U_v verifies whether

  $$e\left(x_v^{-1}W - R, h_1(R)\right) = e\left(Y_s, h_2(m,R)T\right). \quad (6)$$

  If the quality holds, the DV-signature is valid. We show that the verification of Eq. (6) works correctly. From the left-hand side of Eq. (6), we have

  $$\begin{aligned} & e\left(x_v^{-1}W - R, h_1(R)\right) \\ &= e\left(\rho P - R, h_1(R)\right) && \text{(by Eq. (5))} \\ &= e\left(rP + h_2(m,R)\, x_s^{-1}P - R, h_1(R)\right) \\ &= e\left(h_2(m,R)x_s^{-1}P, x_s^2 T\right) && \text{(by Eq. (2))} \\ &= e\left(Y_s, T\right)^{h_2(m,R)} \\ &= e\left(Y_s, h_2(m,R)T\right) \end{aligned}$$

  which leads to the right-hand side of Eq. (6).

4.4. Concrete construction of UDMVS scheme

We introduce the proposed UDMVS scheme in the subsection. In this scheme, all designated verifiers must cooperatively check the validity of received DMV-signature. Since the algorithms of Setup, PSG and PSV are the same to the above, we only describe the others below:

- **DMV-Signature-Generation (DMSG):** Without lost of generality, let $V = \{U_{v_1}, U_{v_2}, \ldots, U_{v_n}\}$ be the group composed of n designated verifiers. To create a DMV-signature for a given message m and its PV-signature $\Omega = (R, T, \rho)$, the designator first chooses $k \in_R Z_p$ to compute

  $$W = \rho P + k\sum_{i=1}^{n} Y_{v_i}, \quad (7)$$

  $$K = kP, \quad (8)$$

  and then deliveries the DMV-signature $\delta = (R, T, W, K)$ along with the corresponding message m to V.
- **DMV-Signature-Verification (DMSV):** Upon receiving (δ, m), $U_{v_i} \in V$ computes

  $$Z_i = x_{v_i} K, \quad (9)$$

  and then sends it to a clerk $U_{ck} \in V$. After collecting all Z_j's, U_{ck} verifies whether

  $$e(W - \sum_{j=1}^{n} Z_j - R, h_1(R)) = (Y_s, h_2(m, R)T). \quad (10)$$

  If it holds, U_{ck} announces the DMV-signature δ for m is valid. We show that the verification of Eq. (10) works correctly. From the left-hand side of Eq. (10), we have

  $$\begin{aligned} & e(W - \sum_{j=1}^{n} Z_j - R, h_1(R)) \\ &= e(W - \sum_{j=1}^{n} x_{v_j} K - R, h_1(R)) && \text{(by Eq. (9))} \\ &= e(W - k\sum_{j=1}^{n} Y_{v_j} - R, h_1(R)) && \text{(by Eq. (8))} \\ &= e(\rho P - R, h_1(R)) && \text{(by Eq. (7))} \\ &= e(rP + h_2(m,R)x_s^{-1}P - R, h_1(R)) \\ &= e(h_2(m,R)x_s^{-1}P, x_s^2 T) && \text{(by Eq. (2))} \\ &= e(Y_s, T)^{h_2(m,R)} \\ &= e(Y_s, h_2(m, R)T) \end{aligned}$$

which leads to the right-hand side of Eq. (10).

5. Security proof and comparison

In this section, we first define the essential security model and then prove the security of our proposed schemes. Some comparisons with related schemes are also made.

5.1. Security Model

A common security requirement of the proposed probabilistic signature scheme and its UDVS/UDMVS extensions is unforgeability against existential forgery under adaptive chosen-message attacks (EF-CMA). According to [19], it is sufficient to prove the unforgeability of DV/DMV-signature, since the property of PV-unforgeability is implied by it. As for a secure UDVS/UDMVS scheme, we have to further consider the security requirement of non-transferability, i.e., the designated verifier/multi-verifier cannot transfer the conviction to any third party. We define these security notions as Definitions 3 and 4.

Definition 3. ***(Strong DV/DMV-Unforgeability)*** *A DV/DMV-signature of the UDVS/UDMVS scheme is said to achieve the security requirement of unforgeability against existential forgery under adaptive chosen-message attacks (EF-CMA) if there is no probabilistic polynomial-time adversary $\mathcal{A}$ with a non-negligible advantage in the following game played with a challenger $\mathcal{B}$:*

Setup: $\mathcal{B}$ first runs the Setup(1^k) algorithm and sends the system's public parameters *params* to the adversary $\mathcal{A}$.

Phase 1: The adversary $\mathcal{A}$ can issue several kinds of queries adaptively, i.e., each query might be based on the result of previous queries:

- *PV-Signature-Generation (PSG) queries:* $\mathcal{A}$ makes a PSG query for a message m. $\mathcal{B}$ returns the corresponding PV-signature Ω.
- *DV/DMV-Signature-Verification (DSV/DMSV) queries:* $\mathcal{A}$ makes a DSV/DMSV query for a pair (δ, m) with respect to the signer and the designated verifier/multi-verifier. $\mathcal{B}$ returns **True** if δ is a valid DV/DMV-signature for m. Otherwise, the error symbol ¶ is returned as a result.

Forgery: Finally, $\mathcal{A}$ produces a new pair (m^*, δ^*) such that the query of PSG (m^*) has never been made. The adversary $\mathcal{A}$ wins if δ^* is a valid DV/DMV-signature for m^*.

Definition 4. ***(Non-Transferability)*** *A UDVS/UDMVS scheme is said to achieve the security requirement of non-transferability if the designated verifier/multi-verifier can simulate a computationally indistinguishable transcript intended for him/them with his/their private key(s).*

5.2. Security proof

We prove that the proposed scheme achieves the above defined security models as Theorems 1 to 4.

Theorem 1. ***(Strong DV-Unforgeability)*** *The DV-signature of our proposed UDVS scheme is (t, q_{h_1}, q_{h_2}, q_{PSG}, q_{DSV}, ε)-secure against existential forgery under adaptive chosen-message attacks (EF-CMA) in the random oracle model if there is no probabilistic polynomial-time adversary $\mathcal{A}$ that can (t', ε')-break the BIDHP, where*

$$\varepsilon' \geq (\varepsilon - 2^{-|\mathbf{G}_1|}),$$

$$t' \approx t + t_\lambda(2q_{DSV}).$$

Here t_λ is the time for performing one bilinear pairing computation.

Proof. Suppose that a probabilistic polynomial-time adversary $\mathcal{A}$ can forge a valid DV-signature of our proposed UDVS scheme with a non-negligible advantage ε under the adaptive chosen message attack after running in time at most t and asking at most q_{h_i} h_i random oracle (for i = 1 and 2), q_{PSG} PSG and q_{DSV} DSV queries. Then we can construct another algorithm $\mathcal{B}$ that (t', ε')-breaks the BIDHP with a non-negligible advantage by taking $\mathcal{A}$ as a subroutine. Let all involved parties and notations be defined the same as those in Section 4.3. The objective of $\mathcal{B}$ is to obtain $e(P, P)^{a^{-1}b}$ by taking (P, aP, bP) as inputs. In this proof, $\mathcal{B}$ simulates a challenger to $\mathcal{A}$ in the following game.

Setup: The challenger $\mathcal{B}$ runs the Setup(1^k) algorithm to obtain the system's public parameters $params = \{\mathbf{G}_1, \mathbf{G}_2, q, P, e\}$. Then $\mathcal{B}$ chooses $d \in_R Z_q$, sets the public keys of the signer U_s and the designated verifier U_v as $Y_s = aP$ and $Y_v = dP$, respectively, and sends ($params$, Y_s, Y_v) to the adversary $\mathcal{A}$.

Phase 1: $\mathcal{A}$ makes the following kinds of queries adaptively:

- h_1 *oracle:* When $\mathcal{A}$ queries an h_1 oracle of $h_1(R)$, $\mathcal{B}$ first checks the h_1_list for a matched entry. Otherwise, $\mathcal{B}$ chooses $c \in_R Z_q$ and adds the entry (R, c, cbP) to the h_1_list. Finally, $\mathcal{B}$ returns cbP as a result.
- h_2 *oracle:* When $\mathcal{A}$ queries an h_2 oracle of $h_2(m, R)$, $\mathcal{B}$ first checks the h_2_list for a matched entry. Otherwise, $\mathcal{B}$ chooses $v_2 \in_R Z_q$ and adds the entry (m, R, v_2) to the h_2_list. Finally, $\mathcal{B}$ returns v_2 as a result.

- *PSG queries:* When $\mathcal{A}$ makes a PSG query for some message m, $\mathcal{B}$ first chooses f, ρ, $v_2 \in_R Z_q$, computes $R = \rho P - f \cdot v_2 \cdot (aP)$, adds the entry ($m$, R, v_2) to the h_2_list, and sets $T = f \cdot h_1(R) = fcbP$. Finally, $\mathcal{B}$ returns $\Omega = (R, T, \rho)$ as the PV-signature for m.
- *DSV queries:* When $\mathcal{A}$ makes a DSV query for some pair (δ, m), $\mathcal{B}$ runs the DSV algorithm with his chosen private key $d \in_R Z_q$ to return the result.

Forgery: At last, $\mathcal{A}$ outputs a forged DV-signature $\delta^* = (R^*, T^*, W^*)$ for his arbitrarily chosen message m^*.

Analysis of the game: For each PSG query, $\mathcal{B}$ returns a valid PV-signature without being accidentally terminated. Moreover, $\mathcal{B}$ answers each h_i random oracle with a computationally indistinguishable value without collision. Let VLD and QH_1 separately be the events that the outputted DV-signature $\delta^* = (R^*, T^*, W^*)$ is valid and $\mathcal{A}$ has ever asks the corresponding $h_1(R^*)$ random oracle. The probability that $\mathcal{A}$ can guess the correct random value without asking the random oracle is not greater than $2^{-|G_1|}$. Since $\mathcal{A}$ has a non-negligible advantage ε to break the proposed scheme under adaptive chosen-message attacks, we have

$$\varepsilon = \Pr[\text{VLD}]$$
$$\le \Pr[\text{VLD} \mid \text{QH}_1] + \Pr[\text{VLD} \mid \neg\text{QH}_1]$$
$$\le \Pr[\text{VLD} \mid \text{QH}_1] + 2^{-|G_1|}.$$
$$\Rightarrow \Pr[\text{VLD} \mid \text{QH}_1] \ge \varepsilon - 2^{-|G_1|}.$$

If the forged DV-signature $\delta^* = (R^*, T^*, W^*)$ for m^* is valid, it will satisfy

$$e(d^{-1}W^* - R^*, h_1(R^*)) = e(Y_s, T^*)^{h_2(m, R^*)}.$$

When the event (VLD | QH_1) occurs, we claim that $T^* = a^{-2}h_1(R^*) = a^{-2}(cbP)$. $\mathcal{B}$ first searches the h_1_list for a match entry (R, c, cbP) where $R = R^*$ and then further computes

$$e(d^{-1}W^* - R^*, h_1(R^*))^{h_2(m, R^*)^{-1}c^{-1}}$$
$$= e(Y_s, T^*)^{h_2(m, R^*)^{-1}c^{-1}h_2(m, R^*)}$$
$$= e(Y_s, T^*)^{c^{-1}}$$
$$= e(aP, a^{-2}(cbP))^{c^{-1}}$$
$$= e(P, P)^{a^{-1}b}.$$

Therefore, we can express the probability of $\mathcal{B}$ to solve the BIDHP as $\varepsilon' \ge (\varepsilon - 2^{-|G_1|})$. The running time required for $\mathcal{B}$ is $t' \approx t + t_\lambda(2q_{DSV})$.

Q.E.D.

Theorem 2. (Strong DMV-Unforgeability) *The DMV-signature of our proposed UDMVS scheme is (t, q_{h_1}, q_{h_2}, q_{PSG}, q_{DMSG}, ε)-secure against existential forgery under adaptive chosen-message attacks (EF-CMA) in the random oracle model if there is no probabilistic polynomial-time adversary $\mathcal{A}_2$ that can (t', ε')-break the BIDHP, where*

$$\varepsilon' \ge (\varepsilon - 2^{-|G_1|}),$$
$$t' \approx t + t_\lambda(2q_{DMSV}).$$

Here t_λ is the time for performing one bilinear pairing computation.

Proof: Suppose that a probabilistic polynomial-time adversary $\mathcal{A}$ can forge a valid DMV-signature of our proposed UDMVS scheme with a non-negligible advantage ε under the adaptive chosen message attack after running in time at most t and asking at most q_{h_i} h_i random oracle (for i = 1 and 2), q_{PSG} PSG and q_{DMSV} DMSV queries. Then we can construct another algorithm $\mathcal{B}$ that (t', ε')-breaks the BIDHP with a non-negligible advantage by taking $\mathcal{A}$ as a subroutine. Let all involved parties and notations be defined the same as those in Section 4.4. The objective of $\mathcal{B}$ is to obtain $e(P, P)^{a^{-1}b}$ by taking (P, aP, bP) as inputs. In this proof, $\mathcal{B}$ simulates a challenger to $\mathcal{A}$ in the following game.

Setup: The challenger $\mathcal{B}$ runs the Setup(1^k) algorithm to obtain the system's public parameters $params = \{\boldsymbol{G}_1, \boldsymbol{G}_2, q, P, e\}$. Then $\mathcal{B}$ chooses $d_1, d_2, \ldots, d_n \in_R Z_q$, sets the public keys of the signer U_s and the designated verifier U_{v_i} as $Y_s = aP$ and $Y_{v_i} = d_iP$, respectively, and sends ($params$, Y_s, Y_{v_i}'s) to the adversary $\mathcal{A}$.

Phase 1: $\mathcal{A}$ adaptively makes queries. For h_1, h_2 and PSG queries, $\mathcal{B}$ responds as those in Theorem 1. When $\mathcal{A}$ makes a DMSV query for some pair (δ, m), $\mathcal{B}$ runs the DMSV algorithm with his chosen private keys $d_1, d_2, \ldots, d_n \in_R Z_q$ to return the result.

Forgery: At last, $\mathcal{A}$ outputs a forged DMV-signature $\delta^* = (R^*, T^*, W^*, K^*)$ for his arbitrarily chosen message m^*.

Analysis of the game: Let VLD and QH_1 be the events defined as those in Theorem 1. According to the analyses of Theorem 1, we can derive that

$$\varepsilon = \Pr[\text{VLD}]$$
$$\le \Pr[\text{VLD} \mid \text{QH}_1] + \Pr[\text{VLD} \mid \neg\text{QH}_1]$$
$$\le \Pr[\text{VLD} \mid \text{QH}_1] + 2^{-|G_1|}.$$
$$\Rightarrow \Pr[\text{VLD} \mid \text{QH}_1] \ge \varepsilon - 2^{-|G_1|}.$$

If the forged DMV-signature $\delta^* = (R^*, T^*, W^*, K^*)$ for m^* is valid, it will satisfy

$$e(W^* - \sum_{i=1}^{n} d_iK^* - R^*, h_1(R^*)) = e(Y_s, T^*)^{h_2(m, R^*)}.$$

When the event (VLD | QH_1) occurs, we claim that $T^* = a^{-2}h_1(R^*) = a^{-2}(cbP)$. $\mathcal{B}$ first searches the h_1_list for a match entry (R, c, cbP) where $R = R^*$ and then further computes

$$e(W^* - \sum_{i=1}^{n} d_i K^* - R^*, h_1(R^*))^{h_2(m, R^*)^{-1}c^{-1}}$$

$$= e(Y_S, T^*)^{h_2(m, R^*)^{-1}c^{-1}h_2(m, R^*)}$$

$$= e(Y_S, T^*)^{c^{-1}}$$

$$= e(aP, a^{-2}(cbP))^{c^{-1}}$$

$$= e(P, P)^{a^{-1}b}.$$

Consequently, we can express the probability of $\mathcal{B}$ to solve the BIDHP as $\varepsilon' \geq (\varepsilon - 2^{-|G_1|})$. The running time required for $\mathcal{B}$ is $t' \approx t + t_\lambda(2q_{DMSV})$.

Q.E.D.

Theorem 3. ***(Non-Transferability)*** *The proposed UDVS scheme satisfies the security requirement of non-transferability.*

Proof: To generate a DV-signature δ^* intended for himself, any designated verifier first chooses $R^* \in_R \boldsymbol{G}_1$ and $f \in_R Z_q$ to compute

$$T^* = f \cdot h_1(R^*), \tag{11}$$

$$W^* = x_V(f \cdot h_2(m, R^*)Y_A + R^*). \tag{12}$$

Here, $\delta^* = (R^*, T^*, W^*)$ is a valid DV-signature for m. The generated δ^* is computationally indistinguishable from the received δ. To be precise, the probability that the computed $\delta^* = (R^*, T^*, W^*)$ and the received $\delta = (R, T, W)$ are identical is at most $2^{-(|G_1| + k)}$, i.e., Pr [$\delta^* = \delta$] $\leq 2^{-(|G_1| + k)}$.

Q.E.D.

Theorem 4. ***(Non-Transferability)*** *The proposed UDMVS scheme satisfies the security requirement of non-transferability.*

Proof: To generate another DMV-signature δ^* intended for the group V, the clerk U_{ck} first chooses R^*, $K^* \in_R \boldsymbol{G}_1$ and then delivers K^* to each $U_{v_i} \in V$. Upon receiving K^*, $U_{v_i} \in V$ computes

$$Z_i^* = x_{v_i} K^*, \tag{13}$$

and then sends it back to U_{ck}. After receiving all Z_j^*'s, U_{ck} chooses $f \in_R Z_q$ and further computes

$$T^* = f \cdot h_1(R^*), \tag{14}$$

$$W^* = f \cdot h_2(m, R^*)Y_S + R^* + \sum_{j=1}^{n} Z_j^*. \tag{15}$$

Here, $\delta^* = (R^*, T^*, W^*, K^*)$ is a valid DMV-signature for m. The generated δ^* is computationally indistinguishable from the received δ. To be precise, the probability that the computed $\delta^* = (R^*, T^*, W^*, K^*)$ and the received $\delta = (R, T, W, K)$ are identical is at most $2^{-(2|G_1| + k)}$, i.e., Pr [$\delta^* = \delta$] $\leq 2^{-(2|G_1| + k)}$.

Q.E.D.

5.3. Comparisons

In this subsection, we compare the proposed schemes with some related ones including Steinfeld *et al.*'s (St-DV for short) [20], the Laguillaumie-Vergnaud (La-DV for short) [28] and Ng *et al.*'s (Ng-DMV for short) [29] schemes in terms of signature type, security assumption and computational efficiency. For convenience, we only evaluate the most time-consuming operations, i.e., the bilinear pairing computation in the following comparisons. Other operations can be ignored, since they are negligible as compared with it. Let T_B be the time for performing one bilinear pairing. The detailed evaluation is demonstrated as Table 1. From the table, it can be seen that the proposed schemes are pairing-free in PSG and DSG/DMSG phases, which benefits to the computation limited devices. Most importantly, the proposed schemes are probabilistic, which means that our schemes have different signatures for the identical message.

Table 1. Comparisons of the proposed and related schemes

Scheme / Item	St-DV	La-DV	Ng-DMV	Ours-DV/DMV
Signature Type	Deterministic			Probabilistic
#Private Key of Each User	1	1	2	1
Security Assumption of PV-signature	co-CDH[1]		BDH	BIDH
#Pairings for PSG	0	0	0	0
#Pairings for PSV	$2T_B$	$2T_B$	$2T_B$	$2T_B$
Security Assumption of DV/DMV-signature	BDH	GBDH[2]	BDH	BIDH
#Pairings for DSG/DMSG	$1T_B$	$1T_B$	$1T_B$	0
#Pairings for DSV/DMSV	$1T_B$	$1T_B$	nT_B[3]	$2T_B$
Security Proof Model	Random Oracle			

Remarks: 1. The term "co-CDH" denotes Computational co-Diffie-Hellman [1].
2. The term "GBDH" denotes Gap-Bilinear Diffie-Hellman [28].
3. n is the number of designated verifiers.

6. Conclusions

In this paper, we proposed a novel and provably secure probabilistic signature scheme and its extensions into UDVS and UDMVS schemes. In a UDVS/UDMVS scheme, the designated verifier/multi-verifier can only check the validity of a DV/DMV-signature, but can not transfer the conviction to any third party, so as to protect the privacy of any signature holder. The underlining security of our proposed signature scheme and its extensions is based on the bilinear inverse Diffie-Hellman problem (BIDHP) which is polynomial-time equivalent to the well-known bilinear Diffie-Hellman problem (BDHP). Compared with previous works, our schemes have crucial benefits to computation constrained devices, as the PV-signature generation and designation are pairing-free. Moreover, we also proved that the proposed schemes achieve the security requirement of unforgeability against existential forgery under adaptive chosen-message attacks (EF-CMA) in the random oracle model.

Acknowledgement

We would like to thank anonymous referees for their valuable suggestions. This work was supported in part by the National Science Council of Republic of China under the contract number NSC 102-2221-E-019-041.

References

[1] **D. Boneh, B. Lynn, H. Shacham.** Short signatures from the Weil pairing. *Journal of Cryptology*, 2004, Vol. 17, No. 4, 297-319.

[2] **W. Diffie, M. Hellman.** New directions in cryptography. *IEEE Transactions on Information Theory*, 1976, Vol. IT-22, No. 6, 644-654.

[3] **T. ElGamal.** A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 1985, Vol. IT-31, No. 4, 469-472.

[4] **R. Rivest, A. Shamir, L. Adleman.** A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 1978, Vol. 21, No. 2, 120-126.

[5] **C. P. Schnorr.** Efficient signature generation by smart cards. *Journal of Cryptology*, 1991, Vol. 4, No. 3, 161-174.

[6] **W. Stallings.** *Cryptography and Network Security: Principles and Practices*. 4th. Ed. Pearson, 2005.

[7] **B. Meng, S. Wang, Q. Xiong.** A fair non-repudiation protocol. In: *Proceedings of the 7th International Conference on Computer Supported Cooperative Work in Design (CSCW'02)*, Brazil, 2002, pp. 68-73.

[8] **I. Ray, N. Narasimhamurthi.** An anonymous electronic voting protocol for voting over the Internet. In: *Proceedings of the 3rd International Workshop on Advanced Issues of E-Commerce and Web-Based Information Systems (WECWIS'01)*, California, 2001, pp. 188-190.

[9] **B. Schoenmakers.** A simple publicly verifiable secret sharing scheme and its application to electronic voting. *Advances in Cryptology – CRYPTO'99*, Springer-Verlag, 1999, 148-164.

[10] **D. Chaum, H. van Antwerpen.** Undeniable signature. *Advances in Cryptology – CRYPTO'89*, Springer-Verlag, 1990, 212-216.

[11] **M. Jakobsson, K. Sako, R. Impagliazzo.** Designated verifier proofs and their applications. *Advances in Cryptology – EUROCRYPT'96*, Springer-Verlag, 1996, 143-154.

[12] **G. Wang.** An Attack on not-interactive designated verifier proofs for undeniable signatures. *Cryptology ePrint archive, Report 2003/243*, 2003. http://eprint.iacr.org/2003/243, 2003.

[13] **S. Saeednia, S. Kremer, O. Markowitch.** An efficient strong designated verifier signature scheme. In: *Proceedings of the 6th International Conference on Information Security and Cryptology (ICISC 2003)*, Seoul, Korea, 2003, pp. 40-54.

[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, Vol. 6, No. 1, 82-93.

[15] **B. Kang, C. Boyd, E. Dawson.** A novel identity-based strong designated verifier signature scheme. *The Journal of Systems and Software*, 2009, Vol. 82, No. 2, 270-273.

[16] **K. Kumar, G. Shailaja, A. Saxena.** Identity based strong designated verifier signature scheme. *Cryptology ePrint Archive, Report 2006/134*, 2006. http://eprint.iacr.org/2006/134.

[17] **W. Susilo, F. Zhang, Y. Mu.** Identity-based strong designated verifier signature schemes. *Information Security and Privacy*, 2004, Vol. 3108, Springer-Verlag, 167-170.

[18] **J. Zhang, J. Mao.** A novel Id-based designated verifier signature scheme. *Information Sciences*, 2008, Vol. 178, No. 3, 766-773.

[19] **E. J. Yoon.** An efficient and secure identity-based strong designated verifier signature scheme. *Information Technology and Control*, 2011, Vol. 40, No. 4, 323-329.

[20] **R. Steinfeld, L. Bull, H. Wang, J. Pieprzyk.** Universal designated-verifier signatures. *Advances in Cryptology – ASIACRYPT'03*, Springer-Verlag, 2003, 523-542.

[21] **R. Steinfeld, H. Wang, J. Pieprzyk.** Efficient extension of standard Schnorr/RSA signatures into universal designated-verifier signatures. In: *Proceedings of Public Key Cryptography (PKC 2004)*, Springer, 2004, pp. 86-100.

[22] **R. Zhang, J. Furukawa, H. Imai.** Short signature and universal designated verifier signature without random oracles. *Applied Cryptography and Network Security (ACNS 2005)*, 2005, Vol. 3531, Springer, 483-498.

[23] **X. Huang, W. Susilo, Y. Mu, W. Wu.** Secure universal designated verifier signature without random oracles. *International Journal of Information Security*, 2008, Vol. 7, No. 3, Springer, 171-183.

[24] **F. Tang, C. Lin, P. Ke.** Universal designated verifier signcryption. *Network and System Security*, LNCS, 2012, Vol. 7645, 126-134.

[25] **P. Thorncharoensri, W. Susilo, Y. Mu.** Universal designated verifier signatures with threshold-signers.

Advances in Information and Computer Security, LNCS, 2009, Vol. 5824, 89-109.

[26] **K. Yoneyama, M. Ushida, K. Ohta.** Rigorous security requirements for designated verifier signatures. *Information Security and Cryptology*, LNCS, 2011, Vol. 6584, 318-335.

[27] **F. Zhang, R. Safavi-Naini, W. Susilo.** An efficient signature scheme from bilinear pairings and its applications. In: *Proceedings of Public Key Cryptography (PKC 2004)*, Springer, 2004, pp. 277-290.

[28] **F. Laguillaumie, D. Vergnaud**. Designated verifier signatures: anonymity and efficient construction from any bilinear map. In: *Proceedings of the 4th Conference on Security in Communication Networks (SCN), LNCS 3352*, Springer, 2005, pp. 105-119.

[29] **C. Y. Ng, W. Susilo, Y. Mu.** Universal designated multi verifier signature schemes. In: *Proceedings of the 11th International Conference on Parallel and Distributed Systems (ICPADS'05)*, 2005, pp. 305-309.

Received February 2013.