Identity-Based Verifiably Encrypted Signatures without Random Oracles

Lei Zhang[1], Qianhong Wu[1,2], and Bo Qin[1,3]

[1] UNESCO Chair in Data Privacy
Department of Computer Engineering and Mathematics
Universitat Rovira i Virgili
Av. Països Catalans 26, E-43007 Tarragona, Catalonia
[2] Key Lab. of Aerospace Information Security and Trusted Computing
Ministry of Education, School of Computer, Wuhan University, China
[3] Dept. of Maths, School of Science, Xi'an University of Technology, China
{lei.zhang,qianhong.wu,bo.qin}@urv.cat

Abstract. Fair exchange protocol plays an important role in electronic commerce in the case of exchanging digital contracts. Verifiably encrypted signatures provide an optimistic solution to these scenarios with an off-line trusted third party. In this paper, we propose an identity-based verifiably encrypted signature scheme. The scheme is non-interactive to generate verifiably encrypted signatures and the resulting encrypted signature consists of only four group elements. Based on the computational Diffie-Hellman assumption, our scheme is proven secure without using random oracles. To the best of our knowledge, this is the first identity-based verifiably encrypted signature scheme provably secure in the standard model.

Keywords: Identity-Based Cryptography, Fair Exchange, Verifiably Encrypted Signature, Provable Security.

1 Introduction

Suppose that two entities Alice and Bob wish to sign some contract. However, neither is willing to produce a signature without being sure that the other will. This kind of applications is referred to as fair exchange. Fair exchange protocols play a critical role in electronic payment systems, contract signing, electronic commerce and certified e-mail. Informally, a fair exchange protocol aims at enabling two entities, who do not fully trust each other, to exchange their items (digital signatures or e-merchandise) in a fair way, so that at the end of the protocol either both entities get the other's item, or neither does, even if the protocol is halted by any reason.

In the last years, a number of fair exchange protocols have been proposed [1,3,9,10,11,15,18,21,23]. Among them, Asokan *et al.* [1] were the first to formally study the problem of optimistic fair exchanges. They introduced a fair exchange protocol relying on a trusted third party (TTP) in an optimistic way, in the

J. Pieprzyk and F. Zhang (Eds.): ProvSec 2009, LNCS 5848, pp. 76–89, 2009.
© Springer-Verlag Berlin Heidelberg 2009

sense that the TTP does not participate in the actual exchange protocol unless one player crashes or attempts to cheat. In this proposal, the TTP is called off-line TTP. However, the protocol by Asokan *et al.* relies on inefficient interactive zero-knowledge proofs.

A verifiably encrypted signature scheme is designed for a fair exchange environment. In this notion, a verifier is able to verify that the encrypted signature hides the user's signature on a publicly agreed message, but the verifier cannot extract the hidden signature. In 2003, Boneh *et al.* [3] proposed the first non-interactive verifiably encrypted signature scheme in traditional certificate-based public key cryptography. They implement their construction by aggregating the BLS short signatures [4]. To realize a fair exchange, Alice sends Bob a verifiably encrypted signature on the contract. Then Bob can send his signature to Alice. If Alice does not respond with her signature, Bob can send to the adjudicator (*i.e.*, the off-line TTP) Alice's verifiably encrypted signature and the transcript of his interaction with Alice. After validating the verifiably encrypted signature and the transcript, the adjudicator will extract Alice's signature and send it to Bob.

Following the work of Boneh *et al.*, several research papers presented other non-interactive verifiably encrypted signature schemes [9,13,15,21,20]. They are all based on traditional public key cryptography, in which a certificate issued by the certificate authority (CA) must be used to provide the binding between an entity and its public key. Such certificate can be verified by anyone and it guarantees the authenticity of an entity's public key. In practice, the management of public key certificates is costly in terms of computation, storage and communication.

To simplify certificate management in traditional public key cryptography, Shamir introduced the concept of ID-based public key cryptography (ID-PKC) [17]. In ID-PKC systems, an entity's public key is directly derived from its public information such as telephone number, e-mail address and IP address. Therefore, there is no need to obtain the certificate to verify the authenticity of the public key, unlike in traditional public key cryptography. Recently, the investigation of ID-based verifiably encrypted signatures (ID-VES) has attracted much attention [6,8,10,11,22,23]. In [8], a generic construction of ID-VES is presented. One may notice that there is a subtlety. In their generic construction, the public key of a user is no longer his identity only, which is different from the traditional definition of ID-PKC. Based on Hess's identity-based signature scheme [12], Kwon and Lee [14] presented an efficient and concise ID-VES scheme of smaller size than previous ID-VES schemes in ID-PKC systems.

1.1 Our Contribution

Most verifiably encrypted signature schemes only provide provable security in the random oracle model [2]. However, security proofs in this model can only be considered as a heuristic argument. Canetti *et al.* [5] demonstrate that behaving like a "random oracle" is not a property that can be realized in general, and that security proofs in the random oracle model do not always imply the security of

the actual scheme in the real world. Therefore, verifiably encrypted signature schemes provably secure without random oracles are of great interest. However, only few verifiably encrypted signature schemes [9,15,21] have been shown secure without using random oracles. These schemes are all presented in the setting of traditional public key cryptography.

In this paper, we propose an identity-based verifiably encrypted signature scheme. The proposed scheme is non-interactive and does not rely on inefficient zero-knowledge proofs. The resulting encrypted signature consists of only four group elements and the scheme is efficient. For security, the proposal is proven secure without relying on random oracles. Furthermore, we use the computational Diffie-Hellman (CDH) assumption. We also exploit the aggregate extraction assumption which is shown equivalent to the CDH assumption [7]. Hence, our scheme enjoys both desirable performance and security, and is practical for fair exchange applications.

We organize the rest of the paper as follows. Section 2 gives some preliminaries, including bilinear maps, computational problems and the notion of ID-VES schemes. Section 3 defines the security of ID-VES schemes. In Section 4 we present the construction of our ID-VES scheme. We prove the security of our scheme in Section 5. Finally, we conclude in Section 6.

2 Preliminaries

In this section, we will review some fundamental background, including the involved computational assumptions and the syntax of identity based verifiably encrypted signatures.

2.1 Bilinear Maps and Computational Problems

Let G_1 be an additive group of prime order q and G_2 be a multiplicative group of the same order. A mapping $e : G_1 \times G_1 \longrightarrow G_2$ is called a bilinear mapping if it satisfies the following properties:

1. Bilinearity: $e(g^a, g^b) = e(g, g)^{ab}$ for all $a, b \in Z_q$.
2. Non-degeneracy: There exists $u, v \in G_1$ such that $e(u, v) \neq 1$.
3. Computability: There exists an efficient algorithm to compute $e(u, v)$ for any $u, v \in G_1$.

Discrete Logarithm (DL) Problem in G_1: Given $h \in G_1$, find an integer $a \in Z_q$ such that $h = g^a$.

Computational Diffie-Hellman (CDH) Problem in G_1: Given (g^α, g^β) for unknown $\alpha, \beta \in Z_q$, compute $g^{\alpha\beta}$.

Aggregate Extraction Problem in G_1: Given $(g^\alpha, g^\beta, g^\gamma, g^\delta, g^{\alpha\gamma+\beta\delta})$ for unknown $\alpha, \beta, \gamma, \delta \in Z_q$, compute $g^{\alpha\gamma}$.

The aggregate extraction problem was introduced by Boneh *et al.* [3]. Coron and Naccache [7] showed that it is equivalent to the standard CDH problem.

2.2 Syntax of Identity-Based Verifiably Encrypted Signature Schemes

An ID-VES scheme is defined by eight algorithms: Setup, AKeyGen, UKeyGen, Sign, Verify, VESign, VEVerify and Adjudication. The description of each algorithm is as follows.

- Setup: This algorithm is performed by the PKG who takes as input a security parameter k to produce a master-key and a list of system parameters params.
- AKeyGen: This algorithm is performed by the adjudicator who takes as input the parameter list params to produce the adjudicator's private and public key pair (x, y).
- UKeyGen: This algorithm takes as input a user's identity ID, the parameter list params and the master-key to produce the user's private key d_{ID}.
- Sign: This algorithm takes as input a message m, an identity ID, the parameter list params and a signer's private key d_{ID} to produce an ordinary signature σ.
- Verify: This algorithm takes as input a message m, a signature σ, the parameter list params, an identity ID to output *True* if the signature is correct, or *False* otherwise.
- VESign: The inputs of this algorithm are the parameter list params, a private key d_{ID} of a signer whose identity is ID, a message m, and the adjudicator's public key y. The output of this algorithm is a verifiably encrypted signature $\overline{\sigma}$ on m.
- VEVerify: This algorithm takes as input a message m, a verifiably encrypted signature $\overline{\sigma}$, the parameter list params, an identity ID and the adjudicator's public key y to output *True* if the signature is correct, or *False* otherwise.
- Adjudication: This algorithm takes as input a message m, a verifiably encrypted signature $\overline{\sigma}$, the parameter list params, an identity ID and the adjudicator's private key x to output the ordinary signature σ on m.

3 Security Definitions of Identity-Based Verifiably Encrypted Signature Schemes

Boneh *et al.* [3] specify two properties (besides correctness) that a verifiably encrypted signature scheme must satisfy: unforgeability and opacity. Unforgeability requires that it be difficult to forge a valid verifiably encrypted signature. Opacity requires that it be difficult, given a verifiably encrypted signature, to extract an ordinary signature on the same message.

The security of ID-VES schemes is defined by following two games which are used to capture unforgeability and opacity properties respectively.

Definition 1. *An ID-VES scheme is said to be existentially unforgeable secure under adaptively chosen message and ID attacks, if no polynomial time adversary has a non-negligible success probability in the following Game 1:*

Game 1 (for unforgeability)

This game is played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$. It has following three stages.

1. $\mathcal{C}$ runs the Setup algorithm, takes as input a security parameter k to obtain the master-key and the system parameter list params. $\mathcal{C}$ also runs AKeyGen to generate the adjudicator's private and public key pair (x, y). Finally, params and y are sent to the adversary $\mathcal{A}$.
2. $\mathcal{A}$ can adaptively query the following oracles:
   - UKeyGen-Oracle: The input of this oracle is an identity ID_i. The output of this oracle is the private key d_i for ID_i.
   - Sign-Oracle: The input of this oracle is an identity ID_i, a message m_i. The output of this oracle is an ordinary signature σ_i. It is required that σ_i be a valid signature on message m_i under identity ID_i.
   - VESign-Oracle: The input of this oracle is an identity ID_i, a message m_i. The output of this oracle is a verifiably encrypted signature $\bar{\sigma}_i$. It is required that $\bar{\sigma}_i$ be a valid verifiably encrypted signature on message m_i under identity ID_i and adjudicator's public key.
   - Adjudication-Oracle: The input of this oracle is an identity ID_i, a message m_i and a verifiably encrypted signature $\overline{\sigma}_i$. The output of this oracle is an ordinary signature σ_i. It is required that σ_i be a valid signature on message m_i under identity ID_i.
3. Eventually, $\mathcal{A}$ outputs a tuple $(ID^*, m^*, \bar{\sigma}^*)$.

We say that $\mathcal{A}$ wins Game 1 if and only if the output $(ID^*, m^*, \bar{\sigma}^*)$ satisfies the following requirements:

- $\bar{\sigma}^*$ is a valid verifiably encrypted signature on message m^* under identity ID^* and adjudicator's public key y.
- ID^* has never been submitted to the UKeyGen-Oracle.
- (ID^*, m^*) has never been submitted to the VESign-Oracle.
- (ID^*, m^*) has never been submitted to the Sign-Oracle.

Definition 2. *An ID-VES scheme is said to be opaque under adaptively chosen message and ID attacks, if no polynomial time adversary has a non-negligible success probability in the following Game 2:*

Game 2 (for opacity)

This game is played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$ which also comprises three stages.

1. $\mathcal{C}$ runs the Setup algorithm, takes as input a security parameter k to obtain master-key and the system parameter list params. $\mathcal{C}$ also runs AKeyGen to generate the adjudicator's private and public key pair (x, y). Finally, $\mathcal{C}$ sends params and y to the adversary $\mathcal{A}$.
2. $\mathcal{A}$ can adaptively query the following oracles as defined in **Game 1:** UKeyGen-Oracle, Sign-Oracle, VESign-Oracle, Adjudication-Oracle.

3. At the end of $\mathcal{A}$'s attack, $\mathcal{A}$ outputs a tuple (ID^*, m^*, σ^*).

We say that $\mathcal{A}$ wins Game 2 if and only if the output (ID^*, m^*, σ^*) satisfies the following requirements:

- σ^* is a valid signature on message m^* under identity ID^*.
- ID^* has not been submitted to the UKeyGen-Oracle.
- (ID^*, m^*) has never been submitted to the Sign-Oracle.
- (ID^*, m^*, λ) has never been submitted to the Adjudication-Oracle, where λ is any valid verifiably encrypted signature on m^* under identity ID^* and adjudicator's public key y.

4 A Concrete Identity-Based Verifiably Encrypted Signature Scheme

In this section, we present the concrete construction of an ID-VES signature. Our scheme is motivated from the Paterson-Schuldt identity-based signature [16] and the Waters's scheme [19]. The construction is as follows:

- Setup: Given a security parameter k, the PKG does the following:
  1. Select a cyclic additive group G_1 which is generated by g with a prime order q, such that there exist a cyclic multiplicative group G_2 of the same order, and a bilinear map $e : G_1 \times G_1 \longrightarrow G_2$.
  2. Randomly choose $s \in Z_q$ and compute $g_1 = g^s$.
  3. Choose cryptographic hash functions $H_1 : \{0,1\}^* \longrightarrow \{0,1\}^l$ and $H_2 : \{0,1\}^* \longrightarrow \{0,1\}^n$.
  4. Randomly select the following elements: $g_2, u', m' \in_R G_1$; $\hat{u}_i \in_R G_1$ for $i = 1, ..., l; \hat{m}_i \in_R G_1$ for $i = 1, ..., n$. Let $\hat{U} = \{\hat{u}_i\}, \hat{M} = \{\hat{m}_i\}$.

  The public parameters are params $= (e, G_1, G_2, g, g_1, g_2, u', \hat{U}, m', \hat{M})$ and the master-key is g_2^s.
- AKeyGen: The adjudicator accepts params, randomly chooses $x \in Z_q$, computes $y = g^x$, and sets (x, y) as the adjudicator's private and public key pair.
- UKeyGen: This algorithm accepts params, master-key g_2^s, an identity $ID \in \{0,1\}^*$ and constructs the private key for the user as follows:
  1. Compute $u = H_1(ID)$, parse u as a bit string $(u_1, ..., u_l) \in \{0,1\}^l$.
  2. Select a random $r_u \in Z_q$.
  3. Output a private key $d_u = (d_0, d_1)$ where

$$d_0 = g_2^s (u' \prod_{i=1}^{l} \hat{u}_i^{u_i})^{r_u}, d_1 = g^{r_u}.$$

- Sign: On input a signer's (whose identity is ID) private key d_u and a message m, the signer performs the following steps:
  1. Compute $u = H_1(ID), w = H_2(m)$ and parse u as a bit string $(u_1, ..., u_l) \in \{0,1\}^l$, and w as a bit string $(w_1, ..., w_n) \in \{0,1\}^n$.

2. Compute $\pi = u' \prod_{i=1}^{l} \hat{u}_i^{u_i}$ (note that this value can be pre-computed by the signer).
3. Select $r_m, r_v \in Z_q$ at random and compute

$$\sigma_1 = d_0 \pi^{r_v} (m' \prod_{i=1}^{n} \hat{m}_i^{w_i})^{r_m}, \sigma_2 = g^{r_m}, \sigma_3 = d_1 g^{r_v}.$$

4. Output $\sigma = (\sigma_1, \sigma_2, \sigma_3)$ as the signer's signature on message m.

– Verify: To verify a signature $\sigma = (\sigma_1, \sigma_2, \sigma_3)$ on a message m signed by a signer with an identity ID, the verifier performs the following steps:
1. Compute $u = H_1(ID), w = H_2(m)$ and parse u as a bit string $(u_1, ..., u_l) \in \{0,1\}^l$, and w as a bit string $(w_1, ..., w_n) \in \{0,1\}^n$.
2. Verify

$$e(\sigma_1, g) \stackrel{?}{=} e(g_1, g_2) e(u' \prod_{i=1}^{l} \hat{u}_i^{u_i}, \sigma_3) e(m' \prod_{i=1}^{n} \hat{m}_i^{w_i}, \sigma_2).$$

If the above equation holds, output *True*. Otherwise, output *False*.

– VESign: On input a signer's (whose identity is ID) private key d_u and a message m, the signer performs the following steps:
1. Compute $u = H_1(ID), w = H_2(m)$ and represent u as a bit string $(u_1, ..., u_l) \in \{0,1\}^l$, and w as a bit string $w = (w_1, ..., w_n) \in \{0,1\}^n$.
2. Compute $\pi = u' \prod_{i=1}^{l} \hat{u}_i^{u_i}$.
3. Select $\bar{r}_u, \bar{r}_m, \bar{r}_v \in Z_q$ and compute

$$\bar{\sigma}_1 = d_0 \pi^{\bar{r}_u} (m' \prod_{i=1}^{n} \hat{m}_i^{w_i})^{\bar{r}_m} y^{\bar{r}_v}, \bar{\sigma}_2 = g^{\bar{r}_m}, \bar{\sigma}_3 = d_1 g^{\bar{r}_u}, \bar{\sigma}_4 = g^{\bar{r}_v}.$$

4. Output $\bar{\sigma} = (\bar{\sigma}_1, \bar{\sigma}_2, \bar{\sigma}_3, \bar{\sigma}_4)$ as the verifiably encrypted signature on message m.

– VEVerify: To verify a verifiably encrypted signature $\bar{\sigma} = (\bar{\sigma}_1, \bar{\sigma}_2, \bar{\sigma}_3, \bar{\sigma}_4)$ on a message m of an identity ID, a verifier performs the following steps:
1. Compute $u = H_1(ID), w = H_2(m)$ and parse u as a bit string $(u_1, ..., u_l) \in \{0,1\}^l$, and w as a bit string $(w_1, ..., w_n) \in \{0,1\}^n$.
2. Verify

$$e(\bar{\sigma}_1, g) \stackrel{?}{=} e(g_1, g_2) e(y, \bar{\sigma}_4) e(u' \prod_{i=1}^{l} \hat{u}_i^{u_i}, \bar{\sigma}_3) e(m' \prod_{i=1}^{n} \hat{m}_i^{w_i}, \bar{\sigma}_2).$$

If the above equation holds, output *True*. Otherwise, output *False*.

– Adjudication: On input a verifiably encrypted signature $\bar{\sigma} = (\bar{\sigma}_1, \bar{\sigma}_2, \bar{\sigma}_3, \bar{\sigma}_4)$ on a message m of an identity ID, the adjudicator performs the following steps:
1. Compute $u = H_1(ID), w = H_2(m)$ and parse u as a bit string $(u_1, ..., u_l) \in \{0,1\}^l$, w as a bit string $(w_1, ..., w_n) \in \{0,1\}^n$.
2. Select $a, b \in Z_q$ at random and compute

$$\sigma_1' = (u' \prod_{i=1}^{l} \hat{u}_i^{u_i})^a (m' \prod_{i=1}^{n} \hat{m}_i^{w_i})^b \bar{\sigma}_1 \bar{\sigma}_4^{-x}, \sigma_2' = \bar{\sigma}_2 g^b, \sigma_3' = \bar{\sigma}_3 g^a.$$

3. Output $\sigma' = (\sigma_1', \sigma_2', \sigma_3')$.

5 Security Analysis

Theorem 1. *If there exists an adversary $\mathcal{A}$ who has an advantage ϵ in forging a verifiably encrypted signature of our ID-VES scheme in an attack modeled by Game 1, within a time span t and querying at most q_e, q_s, q_v and q_a times* UKeyGen-Oracle, Sign-Oracle, VESign-Oracle, *and* Adjudication-Oracle, *respectively; then the CDH problem in G_1 can be solved within time*

$$t + \mathcal{O}(q_e l + (q_s + q_v + q_a)(l + n))\tau_m + \mathcal{O}(q_e + q_s + q_v + q_a)\tau_e$$

with probability

$$\frac{1}{16(q_e + q_s)q_s(l + 1)(n + 1)}\epsilon$$

where τ_m is the time to compute a multiplication in G_1, τ_e is the time to compute a exponentiation in G_1.

Proof. $\mathcal{C}$ is given a CDH problem instance as follows. Given a group G_1, a generator $g \in G_1$, two elements $g^\alpha, g^\beta \in G_1$, $\mathcal{C}$ is asked to output an element $g^{\alpha\beta} \in G_1$. In order to use $\mathcal{A}$ to solve the CDH problem, $\mathcal{C}$ needs to simulate all the oracles in Game 1 for $\mathcal{A}$. $\mathcal{C}$ does it in the following way.

Stage 1. Let $l_u = 2(q_e + q_s)$ and $l_m = 2q_s$. $\mathcal{C}$ randomly selects two integers k_u and k_m such that $0 \leq k_u \leq l$ and $0 \leq k_m \leq n$. We assume that $l_u(l+1) < q$ and $l_m(n+1) < q$ for the given values of q_e, q_s, l and n. It also randomly selects the following integers:

$x' \in Z_{l_u}; z' \in Z_{l_m}; y', w' \in Z_q$; $\hat{x}_1, ..., \hat{x}_l \in Z_{l_u}$; $\hat{y}_1, ..., \hat{y}_l \in Z_q$; $\hat{z}_1, ..., \hat{z}_n \in Z_{l_m}$; $\hat{w}_1, ...\hat{w}_n \in Z_q$.

Let $u = H_1(ID)$ for an identity ID and $w = H_2(m)$ for a message m; parse u as a bit string $(u_1, ..., u_l) \in \{0,1\}^l$, and w as a bit string $(w_1, ..., w_n) \in \{0,1\}^n$. We further define the following functions for u and w as follows:

$$F(u) = x' + \sum_{i=1}^{l} \hat{x}_i u_i - l_u k_u, J(u) = y' + \sum_{i=1}^{l} \hat{y}_i u_i,$$

$$K(w) = z' + \sum_{i=1}^{n} \hat{z}_i w_i - l_m k_m, L(w) = w' + \sum_{i=1}^{n} \hat{w}_i w_i.$$

$\mathcal{C}$ constructs a set of public parameters as follows:

$$\begin{array}{ll} g_1 = g^\alpha & g_2 = g^\beta \\ u' = g_2^{-l_u k_u + x'} g^{y'} & m' = g_2^{-l_m k_m + z'} g^{w'} \\ \text{for } 1 \leq i \leq l, \hat{u}_i = g_2^{\hat{x}_i} g^{\hat{y}_i} & \hat{m}_i = g_2^{\hat{z}_i} g^{\hat{w}_i} \text{ for } 1 \leq i \leq n \end{array}$$

All the above public parameters are passed to $\mathcal{A}$. $\mathcal{C}$ also picks $x \in Z_q$ as adjudicator's private key, sets $y = g^x$ as adjudicator's public key. y is passed to the adversary $\mathcal{A}$ as well.

Note that the master-key will be $g^{\alpha\beta}$ and we have the following equations:

$$u' \prod_{i=1}^{l} \hat{u}_i^{u_i} = g_2^{F(u)} g^{J(u)} \text{ and } m' \prod_{i=1}^{n} \hat{m}_i^{w_i} = g_2^{K(m)} g^{L(m)}.$$

Stage 2. $\mathcal{C}$ simulates all the oracles as follows:

UKeyGen-Oracle: Upon receiving a query for a private key of an identity ID, $\mathcal{C}$ computes $u = H_1(ID)$ and parses u as a bit string $(u_1, ..., u_l) \in \{0,1\}^l$. Although $\mathcal{C}$ does not know the master-key, it still can construct the private key by assuming $F(u) \neq 0 \mod q$. It randomly chooses $r \in Z_q$ and computes the private key as

$$(d_0, d_1) = \left(g_1^{-J(u)/F(u)} (u' \prod_{i=1}^{l} \hat{u}_i^{u_i})^r, g_1^{-1/F(u)} g^r \right).$$

By letting $\hat{r} = r - \alpha/F(u)$, it can be verified that (d_0, d_1) is a valid private key, shown as follows:

$$\begin{aligned}
d_0 &= g_1^{-J(u)/F(u)} (u' \prod_{i=1}^{l} \hat{u}_i^{u_i})^r \\
&= g_1^{-J(u)/F(u)} (g_2^{F(u)} g^{J(u)})^r = g^{-\alpha J(u)/F(u)} (g_2^{F(u)} g^{J(u)})^r \\
&= g^{-\alpha J(u)\ F(u)} (g_2^{F(u)} g^{J(u)})^{\alpha/F(u)} (g_2^{F(u)} g^{J(u)})^{-\alpha/F(u)} (g_2^{F(u)} g^{J(u)})^r \\
&= g^{-\alpha J(u)/F(u)} g^{\alpha\beta} g^{\alpha J(u)/F(u)} (g_2^{F(u)} g^{J(u)})^{\hat{r}} \\
&= g^{\alpha\beta} (g_2^{F(u)} g^{J(u)})^{\hat{r}} = g_2^{\alpha} (g_2^{F(u)} g^{J(u)})^{\hat{r}} = g_2^{\alpha} (u' \prod_{i=1}^{l} \hat{u}_i^{u_i})^{\hat{r}} \\
d_1 &= g_1^{-1/F(u)} g^r = g^{r - \alpha/F(u)} = g^{\hat{r}}
\end{aligned}$$

If $F(u) = 0 \mod q$, since the above computation cannot be performed (division by 0), $\mathcal{C}$ aborts. To make it simple, $\mathcal{C}$ will abort if $F(u) = 0 \mod l_u$. The equivalency can be observed as follows.

The assumption $l_u(l+1) < q$ implies $0 \leq l_u k_u < q$ and $0 \leq x' + \sum_{i=1}^{l} \hat{x}_i u_i < q$. We have $-q < F(u) < q$ which implies if $F(u) = 0 \mod q$ then $F(u) = 0 \mod l_u$. Hence, $F(u) \neq 0 \mod l_u$ implies $F(u) \neq 0 \mod q$. Thus the former condition will be sufficient to ensure that a private key can be computed without abort.

Sign-Oracle: For a given query of a signature on an identity ID and a message m, $\mathcal{C}$ computes $u = H_1(ID), w = H_2(m)$, parses u as a bit string $(u_1, ..., u_l) \in \{0,1\}^l$, and w as a bit string $(w_1, ..., w_n) \in \{0,1\}^n$. If $F(u) \neq 0 \mod l_u$, $\mathcal{C}$ can just construct a private key for ID as in a UKeyGen-Oracle query, and then use the Sign algorithm to create a signature on m. If $F(u) = 0 \mod l_u$, $\mathcal{C}$ will try to construct a signature in a similar way to the construction of a private key in a UKeyGen-Oracle query. Assume $K(w) \neq 0 \mod l_m$. Using the argument mentioned above, this implies $K(w) \neq 0 \mod q$, given the assumption $l_m(n+1) < q$. The signature can be constructed by first randomly selecting $r_u, r_m \in Z_q$

and then computing

$$\begin{aligned}\sigma &= (\sigma_1, \sigma_2, \sigma_3) \\ &= \left((u' \prod_{i=1}^{l} \hat{u}_i^{u_i})^{r_u} g_1^{-L(m)/K(m)} (m' \prod_{i=1}^{n} \hat{m}_i^{w_i})^{r_m}, g_1^{-1/K(m)} g^{r_m}, g^{r_u} \right) \\ &= \left(g_2^{\alpha} (u' \prod_{i=1}^{l} \hat{u}_i^{u_i})^{r_u} (m' \prod_{i=1}^{n} \hat{m}_i^{w_i})^{\hat{r}_m}, g^{\hat{r}_m}, g^{r_u} \right)\end{aligned}$$

where $\hat{r}_m = r_m - \alpha/K(m)$. If $K(m) = 0 \mod l_m$, $\mathcal{C}$ aborts.

VESign-Oracle: On input an identity ID and a message m, let $u = H_1(ID)$ and $w = H_2(m)$, parse u as a bit string $(u_1, ..., u_l) \in \{0,1\}^l$, and w as a bit string $(w_1, ..., w_n) \in \{0,1\}^n$. $\mathcal{C}$ performs the following steps:

1. Submit (ID, m) to the Sign-Oracle; if the Sign-Oracle does not abort, an ordinary signature $\sigma = (\sigma_1, \sigma_2, \sigma_3)$ on m will be returned.
2. Select $\bar{r}_v \in Z_q$, compute $\bar{\sigma}_1 = \sigma_1 y^{\bar{r}_v}, \bar{\sigma}_2 = \sigma_2, \bar{\sigma}_3 = \sigma_3, \bar{\sigma}_4 = g^{\bar{r}_v}$.
3. Output $\bar{\sigma} = (\bar{\sigma}_1, \bar{\sigma}_2, \bar{\sigma}_3, \bar{\sigma}_4)$.

Adjudication-Oracle: On input an identity ID, a message m and a verifiably encrypted signature $\overline{\sigma}$ on m under identity ID, $\mathcal{C}$ can use the Adjudication algorithm of our scheme to produce the ordinary signature, since $\mathcal{C}$ knows the adjudicator's private key.

Stage 3. Finally, $\mathcal{A}$ outputs a verifiably encrypted signature $\bar{\sigma}^* = (\bar{\sigma}_1^*, \bar{\sigma}_2^*, \bar{\sigma}_3^*, \bar{\sigma}_4^*)$ on a message m^* under identity ID^*. Let $u^* = H_1(ID^*), w^* = H_2(m^*)$, represent u^* as $(u_1^*, ..., u_l^*) \in \{0,1\}^l$, w^* as $(w_1^*, ..., w_n^*) \in \{0,1\}^n$; if $F(u^*) \neq 0 \mod q$ or $K(w^*) \neq 0 \mod q$, $\mathcal{C}$ aborts. Otherwise, since $\mathcal{C}$ knows the adjudicator's private key, he can extract the ordinary signature $\sigma^* = (\sigma_1^*, \sigma_2^*, \sigma_3^*)$ on m from $\bar{\sigma}^*$.

By our setting, we have

$$u' \prod_{i=1}^{l} \hat{u}_i^{u_i} = g^{J(u^*)}, m' \prod_{i=1}^{n} \hat{m}_i^{w_i} = g^{K(w^*)}.$$

Since σ^* must satisfy

$$e(\sigma_1^*, g) = e(g_1, g_2) e(u' \prod_{i=1}^{l} \hat{u}_i^{u_i^*}, \sigma_3^*) e(m' \prod_{i=1}^{n} \hat{m}_i^{w_i^*}, \sigma_2^*),$$

we have

$$g^{\alpha\beta} = \frac{\sigma_1^*}{\sigma_3^{*J(u^*)} \sigma_2^{*K(w^*)}}.$$

This is the solution of the CDH Problem.

Let $\ddot{u}_1, ..., \ddot{u}_{q_H}$ be the output of the hash function H_1 appearing in either UKeyGen-Oracle or in Sign-Oracle queries not involving the challenge identity

ID^*, and let $\ddot{m}_1, ..., \ddot{m}_{q_M}$ be the output of the hash function H_2 in the Sign-Oracle. We have $q_H \leq q_e + q_s$ and $q_M \leq q_s$. We also define the events A_i, A^*, B_i, B^* as follows:

$$\begin{aligned}
&A_i : F(\ddot{u}_i) \neq 0 \mod l_u \quad \text{where } i = 1, ..., q_H \\
&A^* : F(u^*) = 0 \mod q \\
&B_i : K(\ddot{m}_j) \neq 0 \mod l_m \text{ where } j = 1, ..., q_M \\
&B^* : K(m^*) = 0 \mod q
\end{aligned}$$

The probability of $\mathcal{C}$ not aborting is:

$$\Pr[\neg \mathsf{abort}] \geq \Pr[\bigwedge_{i=1}^{q_H} A_i \wedge A^* \wedge \bigwedge_{j=1}^{q_M} B_j \wedge B^*].$$

Note that the events $\bigwedge_{i=1}^{q_H} A_i \wedge A^*$ and $\bigwedge_{j=1}^{q_M} B_j \wedge B^*$ are independent. The assumption $l_u(l+1) < q$ implies that if $F(u) = 0 \mod q$ then $F(u) = 0 \mod l_u$. In addition, it also implies that if $F(u) = 0 \mod l_u$, there will be a unique choice of k_u with $0 \leq k_u \leq l$ such that $F(u) = 0 \mod q$. Since k_u, x' and $\hat{x}_1, ..., \hat{x}_l$ are randomly chosen, we have

$$\begin{aligned}
&\Pr[A^*] \\
&= \Pr[F(u^*) = 0 \mod q \wedge F(u^*) = 0 \mod l_u] \\
&= \Pr[F(u^*) = 0 \mod l_u] \Pr[F(u^*) = 0 \mod q | F(u^*) = 0 \mod l_u] \\
&= \frac{1}{l_u} \frac{1}{l+1}
\end{aligned}$$

On the other hand, we have:

$$\begin{aligned}
\Pr[\bigwedge_{i=1}^{q_H} A_i | A^*] &= 1 - \Pr[\bigvee_{i=1}^{q_H} \neg A_i | A^*] \\
&\geq 1 - \sum_{i=1}^{q_H} \Pr[\neg A_i | A^*]
\end{aligned}$$

Note $F(\ddot{u}_i) = 0 \mod l_u$ and $F(\ddot{u}_j) = 0 \mod l_u$ are independent, where $i \neq j$, since the outputs of $F(\ddot{u}_i)$ and $F(\ddot{u}_j)$ will differ in at least one randomly chosen value. Also since the events A_i and A^* are independent for any i, we have $\Pr[\neg A_i | A^*] = 1/l_u$. Hence we have

$$\begin{aligned}
\Pr[\bigwedge_{i=1}^{q_H} A_i \wedge A^*] &= \Pr[A^*] \Pr[\bigwedge_{i=1}^{q_H} A_i | A^*] \\
&\geq \frac{1}{l_u(l+1)} (1 - \frac{q_e + q_s}{l_u})
\end{aligned}$$

and setting $l_u = 2(q_e + q_s)$ as in the simulation gives

$$\Pr[\bigwedge_{i=1}^{q_H} A_i \wedge A^*] \geq \frac{1}{4(q_e + q_s)(l+1)}.$$

A similar analysis for the Sign-Oracle queries gives the result

$$\Pr[\bigwedge_{i=1}^{q_M} B_j \wedge B^*] \geq \frac{1}{4q_s(n+1)}.$$

And we get that

$$\begin{aligned}\Pr[\neg\mathsf{abort}] &\geq \Pr[\bigwedge_{i=1}^{q_H} A_i \wedge A^*] \Pr[\bigwedge_{j=1}^{q_M} B_j \wedge B^*] \\ &\geq \frac{1}{16(q_e+q_s)q_s(l+1)(n+1)}.\end{aligned}$$

If the simulation does not abort, $\mathcal{A}$ will create a valid forgery with probability at least ϵ. Algorithm $\mathcal{C}$ can then compute $g^{\alpha\beta}$ from the forgery as shown above.

The time complexity of algorithm $\mathcal{C}$ is dominated by the exponentiations and, for larger values of l and n, by the multiplications performed in the UKeyGen-Oracle, Sign-Oracle, VESign-Oracle, and Adjudication-Oracle queries. For UKeyGen-Oracle, Sign-Oracle, VESign-Oracle, and Adjudication-Oracle, there are $\mathcal{O}(l)$, $\mathcal{O}(l+n)$, $\mathcal{O}(l+n)$ and $\mathcal{O}(l+n)$ multiplications in G_1 and $\mathcal{O}(1)$, $\mathcal{O}(1)$, $\mathcal{O}(1)$ and $\mathcal{O}(1)$ exponentiations in G_1. Hence, the time complexity of $\mathcal{C}$ is $t + \mathcal{O}(q_e l + (q_s + q_v + q_a)(l+n))\tau_m + \mathcal{O}(q_e + q_s + q_v + q_a)\tau_e$. Thus, the theorem follows.

Theorem 2. *If there exists an adversary $\mathcal{A}$ who has an advantage ϵ in an attack modeled by Game 2, within a time span t and querying at most q_e, q_s, q_v and q_a times* UKeyGen-Oracle, Sign-Oracle, VESign-Oracle, *and* Adjudication-Oracle, *respectively; then the aggregate extraction problem in G_1 can be solved within time*

$$t + \mathcal{O}(q_e l + (q_s + q_v + q_a)(l+n))\tau_m + \mathcal{O}(q_e + q_s + q_v + q_a)\tau_e$$

and with probability

$$\frac{1}{16(q_e+q_s)q_s(l+1)(n+1)}\epsilon.$$

Due to page limitation, the proof will be presented in the full version of this paper.

6 Conclusion

We presented an identity-based verifiably encrypted signature. The proposal is proven existentially unforgeable and opaque against adaptively chosen-message attack assuming that the CDH problem (and its equivalent variant of the aggregate extraction problem) is hard. Unlike existing schemes, our construction does not rely on the existence of random oracles. Our ID-VES scheme is also efficient and can be efficiently applied to fair exchange protocols.

Acknowledgments and Disclaimer

Thanks go to Josep Domingo-Ferrer for his comments on a draft of this paper. The authors are partly supported by the Spanish Government through projects TSI2007-65406-C03-01 "E-AEGIS" and CONSOLIDER INGENIO 2010 CSD2007-00004 "ARES", and by the Government of Catalonia under grant 2009 SGR 1135. They are with the UNESCO Chair in Data Privacy, but their views do not necessarily reflect the position of UNESCO nor commit that organization.

References

1. Asokan, N., Shoup, V., Waidner, M.: Optimistic fair exchange of digital signatures. IEEE Journal on Selected Areas in Communications 18(4), 593–610 (2000)
2. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: ACM CCCS 1993, pp. 62–73. ACM Press, New York (1993)
3. Boneh, D., Gentry, C., Lynn, B., Shacham, H.: Aggregate and verifiably encrypted signatures from bilinear maps. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 416–432. Springer, Heidelberg (2003)
4. Boneh, D., Lynn, B., Shacham, H.: Short signatures from the weil pairing. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 514–532. Springer, Heidelberg (2001)
5. Canetti, R., Goldreich, O., Halevi, S.: The random oracle methodology, revisited. J. ACM 51(4), 557–594 (2004)
6. Cheng, X., Liu, J., Wang, X.: Identity-based aggregate and verifiably encrypted signatures from bilinear pairing. In: Gervasi, O., Gavrilova, M.L., Kumar, V., Laganá, A., Lee, H.P., Mun, Y., Taniar, D., Tan, C.J.K. (eds.) ICCSA 2005. LNCS, vol. 3483, pp. 1046–1054. Springer, Heidelberg (2005)
7. Coron, J.-S., Naccache, D.: Boneh *et al.*'s k-element aggregate extraction assumption is equivalent to the diffie-hellman assumption. In: Laih, C.-S. (ed.) ASIACRYPT 2003. LNCS, vol. 2894, pp. 392–397. Springer, Heidelberg (2003)
8. Galindo, D., Herranz, J., Kiltz, E.: On the generic construction of identity-based signatures with additional properties. In: Lai, X., Chen, K. (eds.) ASIACRYPT 2006. LNCS, vol. 4284, pp. 178–193. Springer, Heidelberg (2006)
9. Gorantla, M., Saxena, A.: Verifiably encrypted signature without random oracles. In: Chakraborty, G. (ed.) ICDCIT 2005. LNCS, vol. 3816, pp. 357–363. Springer, Heidelberg (2005)
10. Gu, C., Zhu, Y.: An ID-based verifiably encrypted signature scheme based on Hess's scheme. In: Feng, D., Lin, D., Yung, M. (eds.) CISC 2005. LNCS, vol. 3822, pp. 42–52. Springer, Heidelberg (2005)
11. Gu, C., Zhu, Y., Zhang, Y.: An ID-based optimistic fair signature exchange protocol from pairings. In: Hao, Y., Liu, J., Wang, Y.-P., Cheung, Y.-m., Yin, H., Jiao, L., Ma, J., Jiao, Y.-C. (eds.) CIS 2005. LNCS, vol. 3802, pp. 9–16. Springer, Heidelberg (2005)
12. Hess, F.: Efficient identity based signature schemes based on pairings. In: Nyberg, K., Heys, H.M. (eds.) SAC 2002. LNCS, vol. 2595, pp. 310–324. Springer, Heidelberg (2003)
13. Hess, F.: On the security of the verifiably-encrypted signature scheme of Boneh, Gentry, Lynn and Shacham. Information Processing Letters 89(3), 111–114 (2004)

14. Kwon, S., Lee, S.H.: An efficient ID-based verifiably encrypted signature scheme based on Hess's scheme. In: Dawson, E., Wong, D.S. (eds.) ISPEC 2007. LNCS, vol. 4464, pp. 93–104. Springer, Heidelberg (2007)
15. Lu, S., Ostrovsky, R., Sahai, A., Shacham, H., Waters, B.: Sequential aggregate signatures and multisignatures without random oracles. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 465–485. Springer, Heidelberg (2006)
16. Paterson, K., Schuldt, J.: Efficient identity-based signatures secure in the standard model. In: Batten, L.M., Safavi-Naini, R. (eds.) ACISP 2006. LNCS, vol. 4058, pp. 207–222. Springer, Heidelberg (2006)
17. Shamir, A.: Identity-based cryptosystems and signature schemes. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (1985)
18. Shao, Z.: Certificate-based verifiably encrypted signatures from pairings. Information Sciences 178(10), 2360–2373 (2008)
19. Waters, B.: Efficient identity-based encryption without random oracles. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 114–127. Springer, Heidelberg (2005)
20. Zhang, F., Safavi-Naini, R., Susilo, W.: Efficient verifiably encrypted signature and partially blind signature from bilinear pairings. In: Johansson, T., Maitra, S. (eds.) INDOCRYPT 2003. LNCS, vol. 2904, pp. 191–204. Springer, Heidelberg (2003)
21. Zhang, J., Mao, J.: A novel verifiably encrypted signature scheme without random oracle. In: Dawson, E., Wong, D.S. (eds.) ISPEC 2007. LNCS, vol. 4464, pp. 65–78. Springer, Heidelberg (2007)
22. Zhang, J., Zou, W.: A robust verifiably encrypted signature scheme. In: Zhou, X., Sokolsky, O., Yan, L., Jung, E.-S., Shao, Z., Mu, Y., Lee, D.C., Kim, D.Y., Jeong, Y.-S., Xu, C.-Z. (eds.) EUC Workshops 2006. LNCS, vol. 4097, pp. 731–740. Springer, Heidelberg (2006)
23. Zhang, Z., Feng, D., Xu, J., Zhou, Y.: Efficient ID-based optimistic fair exchange with provable security. In: Qing, S., Mao, W., López, J., Wang, G. (eds.) ICICS 2005. LNCS, vol. 3783, pp. 14–26. Springer, Heidelberg (2005)