# (Convertible) Undeniable Signatures without Random Oracles

Tsz Hon Yuen[1], Man Ho Au[1], Joseph K. Liu[2], and Willy Susilo[1]

[1] Centre for Computer and Information Security Research
School of Computer Science and Software Engineering
University of Wollongong
Wollongong, Australia
{thy738, aau, wsusilo}@uow.edu.au
[2] Institute for Infocomm Research
Singapore
ksliu@i2r.a-star.edu.sg

**Abstract.** We propose a convertible undeniable signature scheme without random oracles. Our construction is based on the Waters signatures proposed in Eurocrypt 2005. The security of our scheme is based on the CDH and the decision linear assumption. Comparing only the part of undeniable signatures, our scheme uses more standard assumptions than the existing undeniable signatures without random oracles due to Laguillaumie and Vergnaud.

Keywords: Convertible undeniable signature, pairings

## 1 Introduction

Standard digital signatures allow universal verification. However in some real world scenarios, privacy is an important issue. In this situation, we may require that the verification of signatures is restricted by the signer. Then, the verification of a signature requires an interaction with the signer. A signer can deny generating a signature that he never signs, but he cannot deny one that he signs. The proof by the signer cannot be transferred to convince other verifiers. This concept is known as the "Undeniable Signatures" that was proposed by Chaum and van Antwerpen [12]. Later, Boyar, Chaum, Damgård and Pedersen [7] proposed an extension called "Convertible Undeniable Signatures", which allows the possibility to transform an undeniable signature into a self-authenticating signature. This transformation can be restricted to a particular signature only, or can be applied to all signatures of a signer.

There are many different undeniable signatures with variable features and security levels. These features include convertibility [7, 14, 29, 30], designated verifier technique [21], designated confirmer technique [11, 33], identity based scheme [28], time-selective scheme [27], etc. The security for undeniable signatures is said to be *secure* if it is unforgeable, invisible and the confirmation and disavowal protocols are zero-knowledge. It is believed that the zero-knowledgeness is required to make undeniable signatures non-transferable. However, Kurosawa and Heng [24] suggested that zero-knowledgeness and non-transferability can be separated; and the concept of witness indistinguishability can be incorporated. They proposed another security notion called impersonation attack.

The random oracle model [3] is a popular technique in provable security. However several papers proved that some cryptosystems secure in the random oracle were actually provably insecure when the random oracle was instantiated by any real-world hashing functions [10, 2]. As a result, recently there are many new signature schemes which prove their security without random oracles, such as group signatures [1, 9], ring signatures [13, 5], blind signatures [22], group-oriented signatures [36], undeniable signatures [26], universal designated verifier signatures [39], etc. Nonetheless, some of them introduce new security assumptions that are not well studied, which are the main drawback of some schemes.

**Our Contribution.** We propose the *first* convertible undeniable signatures without random oracles in pairings. Most of the existing convertible undeniable signatures are proven secure in the random oracle model only [7, 29–31, 27] [3], except the recent construction in RSA [25].

Most efficient undeniable signatures are proven secure in the random oracle model only. [15] is secure in the random oracle model currently. [4] Recently, Laguillaumie and Vergnaud proposed the first efficient undeniable signatures without random oracles [26]. However, their anonymity relies on their *new assumption* DSDH, while their unforgeability relies on the GSDH assumption with the access of a DSDH oracle, which seems to be contradictory. Our proposed variant of undeniable signature is proven unforgeable by the CDH assumption and anonymous by the decision linear assumption. Therefore by removing the protocol for convertible parts, our undeniable signature scheme is the *first* proven secure scheme *without using random oracles* and *without using a new assumption* in discrete logarithm settings.

**Recent Works.** An earlier version of the scheme in this section appears in [38]. In 2007, Huang *et al.* [20] proposed a pairing-based convertible undeniable signatures secure in the random oracle model. Huang *et al.* [19] also proposed a generic construction of universally-convertible undeniable signatures from a strongly unforgeable classic signature scheme, a selectively-convertible undeniable signature scheme and a collision resistant hash function. In 2008, Kurosawa and Furukawa [23] defined the universal composability security of undeniable signatures.

In 2009, Phong *et al.* [35] proposed a new RSA-based selectively-convertible undeniable signatures. They also demonstrated an attack on the invisibility of the RSA-based construction in [25]. Phong *et al.* [34] proposed a new discrete-logarithm based selectively-convertible undeniable signature. This scheme is more efficient than our scheme proposed in this section. They pointed out a flaw in the earlier version of our scheme in [38]. This problem is fixed in the proposed scheme in this paper.

**Organization.** The next section briefly explains the pairings and some related intractability problems. Section 3 gives the security model. Section 4 gives our construction and security proofs. The paper ends with some concluding remarks.

## 2 Preliminaries

### 2.1 Pairings and Intractability Problem

Our scheme uses bilinear pairings on elliptic curves. We now give a brief revision on the property of pairings and some candidate hard problems from pairings that will be used later.

Let $\mathbb{G}, \mathbb{G}_T$ be cyclic groups of prime order $p$, writing the group action multiplicatively. Let $g$ be a generator of $\mathbb{G}$.

**Definition 1.** *A map $\hat{e} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ is called a bilinear pairing if, for all $x, y \in \mathbb{G}$ and $a, b \in \mathbb{Z}_p$, we have $\hat{e}(x^a, y^b) = \hat{e}(x, y)^{ab}$, and $\hat{e}(g, g) \neq 1$.*

**Definition 2 (CDH).** *The Computational Diffie-Hellman (CDH) problem is that, given $g$, $g^x, g^y \in \mathbb{G}$ for unknown $x, y \in \mathbb{Z}_p^*$, to compute $g^{xy}$.*

We say that the $(\epsilon, t)$-CDH assumption holds in $\mathbb{G}$ if no $t$-time algorithm has the non-negligible probability $\epsilon$ in solving the CDH problem.

**Definition 3 (Decision Linear [6]).** *The Decision Linear problem is that, given $u$, $u^a$, $v$, $v^b$, $h$, $h^c \in \mathbb{G}$ for unknown $a, b, c \in \mathbb{Z}_p^*$, to output 1 if $c = a + b$ and output 0 otherwise.*

[3] [14] does not prove the invisibility property. The authors only conjecture the security in section 5.1 and 5.2.

[4] Refer to section 1.1 in [25] for details.

We say that the $(\epsilon, t)$-Decision Linear assumption holds in $\mathbb{G}$ if no $t$-time algorithm has probability over half $\epsilon$ in solving the Decision Linear problem in $\mathbb{G}$. The decision linear assumption is proposed in [6] to prove the security of short group signatures. It is also used in [8] and [18] for proving the security of anonymous hierarchical identity-based encryption and obfuscating re-encryption respectively.

## 3 Security Models of Undeniable Signatures

In this section we review the security notions and model of (convertible) undeniable signatures. Unforgeability and invisibility are popular security requirement for undeniable signatures. Kurosawa and Heng [24] proposed another security notion called impersonation. We will use the security model of [24], and extend it to convertible undeniable signatures. The changes for convertible undeniable signatures will be given in brackets.

### 3.1 Security Notions

An (convertible) undeniable signature scheme has the following algorithms:

- **Setup**$(1^\lambda)$: the setup algorithm takes a unary security parameter $\lambda$ as input, and outputs some public parameters $\mathsf{param}$.
- **KeyGen**$(\mathsf{param})$: the key generation algorithm takes the public parameters $\mathsf{param}$ as input, and outputs a public key $\mathsf{pk}$ and a secret key $\mathsf{sk}$.
- **USign**$(\mathsf{param}, \mathsf{sk}, m)$: the signing algorithm takes the public parameters $\mathsf{param}$, a secret key $\mathsf{sk}$ and a message $m$ as inputs, and outputs an undeniable signature $\sigma$.
- **Confirm/Deny.** This is an interactive protocol between a prover and a verifier. Their common inputs are the public parameters $\mathsf{param}$, a public key $\mathsf{pk}$, a message $m$ and a signature $\sigma$. The prover's private input is a secret key $\mathsf{sk}$. At the end of the protocol, the verifier outputs 1 if $\sigma$ is a valid signature of $m$ and outputs 0 otherwise.

  (The following algorithms are for convertible schemes only.)
- **IConvert**$(\mathsf{param}, \mathsf{sk}, m, \sigma)$: The individual conversion algorithm takes the public parameters $\mathsf{param}$, a secret key $\mathsf{sk}$, a message $m$ and a signature $\sigma$ as inputs, and outputs an individual receipt $r$ which makes it possible to individually verify $\sigma$.
- **IVerify**$(\mathsf{param}, \mathsf{pk}, m, \sigma, r)$: The individual verification algorithm takes the public parameters $\mathsf{param}$, a public key $\mathsf{pk}$, a message $m$, a signature $\sigma$ and an individual receipt $r$ as inputs, and
  - outputs $\perp$ if $r$ is an invalid individual receipt, or
  - outputs 1 if $\sigma$ is a valid signature of $m$, or
  - outputs 0 if $\sigma$ is not a valid signature of $m$.
- **UConvert**$(\mathsf{param}, \mathsf{sk})$: The universal conversion algorithm takes the public parameters $\mathsf{param}$ and a secret key $\mathsf{sk}$ as inputs, and outputs an universal receipt $R$ which makes it possible to universally verify all signatures for $\mathsf{pk}$.
- **UVerify**$(\mathsf{param}, \mathsf{pk}, m, \sigma, R)$: The universal verification algorithm takes the public parameters $\mathsf{param}$, a public key $\mathsf{pk}$, a message $m$, a signature $\sigma$ and an universal receipt $R$ as inputs, and
  - outputs $\perp$ if $R$ is an invalid universal receipt, or
  - outputs 1 if $\sigma$ is a valid signature of $m$, or
  - outputs 0 if $\sigma$ is not a valid signature of $m$.

The convertible undeniable signature schemes with all four algorithms (**IConvert, IVerify, UConvert, UVerify**) are sometimes denoted as *universally-convertible undeniable signature*. The convertible undeniable signature schemes with only the algorithms (**IConvert, IVerify**) are sometimes denoted as *selectively-convertible undeniable signature*.

### 3.2 Unforgeability

Strong unforgeability against chosen message attack is defined as in the following game involving an adversary $\mathcal{A}$ and a challenger over message space $\mathcal{M}$.

1. The challenger runs the algorithm $\mathsf{param} \leftarrow \mathbf{Setup}(1^\lambda)$ and $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathbf{KeyGen}(\mathsf{param})$. The challenger gives $\mathsf{param}$ and $\mathsf{pk}$ to $\mathcal{A}$. (For convertible schemes, the challenger also gives $\mathcal{A}$ the universal receipt $R \leftarrow \mathbf{UConvert}(\mathsf{param}, \mathsf{sk})$.)
2. $\mathcal{A}$ can query the following oracles adaptively:
   - Signing oracle: $\mathcal{A}$ requests a signature on any message $m \in \mathcal{M}$ and the challenger responds with $\sigma \leftarrow \mathbf{USign}(\mathsf{param}, \mathsf{sk}, m)$.
   - Confirmation/disavowal oracle: $\mathcal{A}$ queries the oracle with input message-signature pair $(m, \sigma)$. If it is a valid pair, the challenger returns a bit $\mu = 1$ and proceeds with the execution of the **Confirm** protocol with $\mathcal{A}$. Otherwise, the challenger returns a bit $\mu = 0$ and proceeds with the execution of the **Deny** protocol with $\mathcal{A}$.
     (For convertible scheme, this oracle is not necessary as the universal receipt is given.)
3. Finally $\mathcal{A}$ outputs a message-signature pair $(m^*, \sigma^*)$.

$\mathcal{A}$ wins the game if $\sigma^*$ is a valid signature for $m^*$ and the pair $(m^*, \sigma^*)$ is not the output from the signing oracle.

**Definition 4.** *An (convertible) undeniable signature scheme is $(\epsilon,\ t,\ q_c,\ q_s)$-strongly unforgeable against chosen message attack if there is no $t$ time adversary winning the above game with probability greater than $\epsilon$, where $q_c$ and $q_s$ are the number of queries to the confirmation/disavowal oracle and the signing oracle respectively.*

### 3.3 Invisibility

Invisibility against chosen message attack is defined as in the following game involving an adversary $\mathcal{A}$ and a challenger over message space $\mathcal{M}$.

1. The challenger runs the algorithm $\mathsf{param} \leftarrow \mathbf{Setup}(1^\lambda)$ and $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathbf{KeyGen}(\mathsf{param})$. The challenger gives $\mathsf{param}$ and $\mathsf{pk}$ to $\mathcal{A}$.
2. $\mathcal{A}$ can query the following oracles adaptively:
   - Signing oracle and Confirmation/disavowal oracle: they are the same as that in the unforgeability game.
   - (For convertible schemes only.) Receipt generating oracle: $\mathcal{A}$ queries the oracle with input message-signature pair $(m, \sigma)$, and the challenger returns an individual receipt $r$.
3. $\mathcal{A}$ outputs a message $m^*$. The challenger choose a random bit $b^*$. If $b^* = 1$, then $\sigma^* \leftarrow \mathbf{USign}(\mathsf{param}, \mathsf{sk}, m^*)$. Otherwise $\sigma^*$ is chosen uniformly at random from the signature space of the scheme.
4. $\mathcal{A}$ can adaptively query the signing oracle and confirmation/disavowal oracle, where no signing query (and receipt generating query) for $m^*$ and no confirmation/disavowal query for $(m^*, \sigma^*)$ is allowed.
5. Finally $\mathcal{A}$ outputs a guessing bit $b'$

$\mathcal{A}$ wins the game if $b^* = b'$ and there is no confirmation/disavowal query (and receipt generating query) for $(m^*, \sigma^*)$. $\mathcal{A}$'s advantage is $Adv(\mathcal{A}) = |\Pr[b' = b^*] - \frac{1}{2}|$.

**Definition 5.** *An (convertible) undeniable signature scheme is $(\epsilon,\ t,\ q_c,\ q_r,\ q_s)$-invisible if there is no $t$ time adversary winning the above game with advantage greater than $\epsilon$, where $q_c$, $(q_r)$ and $q_s$ are the number of queries to the confirmation/disavowal oracle, (the receipt generating oracle) and the signing oracle respectively..*

### 3.4 Impersonation

Impersonation against chosen message attack is defined as in the following game involving an adversary $\mathcal{A}$ and a challenger over message space $\mathcal{M}$.

1. The challenger runs the algorithm $\mathsf{param} \leftarrow \mathbf{Setup}(1^\lambda)$ and $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathbf{KeyGen}(\mathsf{param})$. The challenger gives $\mathsf{param}$ and $\mathsf{pk}$ to $\mathcal{A}$.
2. $\mathcal{A}$ can query the Signing oracle and the Confirmation/disavowal oracle, which are the same as the one in the unforgeability game.
3. Finally $\mathcal{A}$ outputs a message-signature pair $(m^*, \sigma^*)$ and a bit $b^*$. If $b^* = 1$, $\mathcal{A}$ executes the confirmation protocol with the challenger. Otherwise, $\mathcal{A}$ executes the disavowal protocol with the challenger.

$\mathcal{A}$ wins the game if the challenger is convinced that $\sigma^*$ is a valid signature for $m^*$ if $b^* = 1$, or is an invalid signature for $m^*$ if $b^* = 0$.

**Definition 6.** *An (convertible) undeniable signature scheme is $(\epsilon, t, q_c, q_s)$-secure against impersonation if there is no $t$ time adversary winning the above game with probability at least $\epsilon$, where $q_c$ and $q_s$ are the number of queries to the confirmation/disavowal oracle and the signing oracle respectively.*

*Remark.* For convertible schemes, if an adversary can forge an individual or universal receipt, he can always convince a verifier in the interactive protocol, by directly giving the receipt to him. Therefore the model of impersonation attack already includes the security notion regarding receipts in convertible schemes.

## 4 Convertible Undeniable Signature Scheme

An earlier version of our scheme in [38] used the Waters signatures [37] and the 3-move witness indistinguishable protocol by Kurosawa and Heng [24]. However, Ogata *et al.* [32] later showed that any 3-move confirmation/disavowal protocols are not secure against active attacks. As a result, the 3-move protocol by Kurosawa and Heng is insecure. Therefore, we propose the use of the standard 4-move proof of knowledge of discrete logarithm, or the non-interactive zero-knowledge proof system for bilinear groups by Groth and Sahai [16], to replace the protocol by Kurosawa and Heng in [38]. On the other hand, we use the generic construction of strongly unforgeable signatures in [4] to solve the security problem mentioned in [34]. We also use the proof technique in [17] to achieve a tight security reduction.

### 4.1 Scheme Construction

In this section, we present our convertible undeniable signature scheme. The scheme consists of the following algorithms.

- $\mathbf{Setup}(1^\lambda)$. Let $\mathbb{G}, \mathbb{G}_T$ be groups of prime order $p$. Select generators $g, g_2 \in \mathbb{G}$. Generator $u'$ $\in \mathbb{G}$ is selected in random, and a random $n$-length vector $\mathsf{U} = (u_i)$, whose elements are chosen at random from $\mathbb{G}$. Select an integer $\ell$ as a system parameter. Let $H : \{0,1\}^n \rightarrow \mathbb{Z}_\ell^*$ be a collision resistant hash function. Let $\mathsf{SIG}_{OT} = (\mathsf{Kg}_{OT}, \mathsf{Sign}_{OT}, \mathsf{Verify}_{OT})$ be a secure one time signature scheme and the length of the verification key $vk_{OT}$ is $n$-bits. The system parameters $\mathsf{param}$ are
$$(g, g_2, u', \mathsf{U}, H).$$
- $\mathbf{KeyGen}(\mathsf{param})$. Randomly select $\alpha, \beta', \beta_i \in \mathbb{Z}_p^*$ for $1 \leq i \leq \ell$. Set $g_1 = g^\alpha$, $v' = g^{\beta'}$ and $v_i = g^{\beta_i}$. The public keys $\mathsf{pk}$ are $(g_1, v', v_1, \ldots, v_\ell)$. The secret keys $\mathsf{sk}$ are $(\alpha, \beta', \beta_1, \ldots, \beta_\ell)$.

– **USign**($\mathsf{param}, \mathsf{sk}, m$). To sign a message $m$, the signer runs $(sk_{OT}, vk_{OT}) \leftarrow \mathsf{Kg}_{OT}(1^\lambda)$. Denote $vk_{OT} = (vk_1, \ldots, vk_n) \in \{0,1\}^n$, and denote $\bar{vk} = H(vk_{OT})$. The signer picks $r \in_R \mathbb{Z}_p^*$ and computes the signature

$$S_1 = g_2^\alpha (u' \prod_{i=1}^{n} u_i^{vk_i})^r, \qquad S_2 = (v' \prod_{i=1}^{\ell} v_i^{\bar{vk}^i})^r, \qquad S_3 = \mathsf{Sign}_{OT}(sk_{OT}, m||S_1||S_2).$$

The output signature $\sigma$ is $(S_1, S_2, S_3, vk_{OT})$.

– **Confirm/Deny.** On input a signature $\sigma = (S_1, S_2, S_3, vk_{OT})$, the signer computes:

$$\begin{aligned} L &= \hat{e}(g, g_2), \\ M &= \hat{e}(g_1, g_2), \\ N &= \hat{e}(v' \prod_{i=1}^{\ell} v_i^{\bar{vk}^i}, g_2), \\ O &= \hat{e}(v' \prod_{i=1}^{\ell} v_i^{\bar{vk}^i}, S_1) / \hat{e}(S_2, u' \prod_{i=1}^{n} u_i^{vk_i}). \end{aligned} \tag{1}$$

Note that $\alpha = \log_L M$ and $\log_N O$. The zero-knowledge proof of knowledge can be implemented using known 4-move proof of knowledge of discrete logarithm, or the non-interactive zero-knowledge proof system for bilinear groups by Groth and Sahai [16].

– **IConvert**($\mathsf{param}, \mathsf{sk}, m, \sigma$). Upon input the signature $\sigma = (S_1, S_2, S_3, vk_{OT})$ on the message $m$, the signer computes $\bar{vk} = H(vk_{OT})$ and

$$S_2' = S_2^{1/(\beta' + \sum_{i=1}^{\ell} \beta_i \bar{vk}^i)}.$$

The signer outputs the individual receipt $r = S_2'$ for message $m$.

– **IVerify**($\mathsf{param}, \mathsf{pk}, m, \sigma, r$). Upon input the signature $\sigma = (S_1, S_2, S_3, vk_{OT})$ for the message $m$ and the individual receipt $r = S_2'$, compute $\bar{vk} = H(vk_{OT})$ and check if:

$$\hat{e}(g, S_2) \stackrel{?}{=} \hat{e}(S_2', v' \prod_{i=1}^{\ell} v_i^{\bar{vk}^i}).$$

If they are not equal, output $\perp$. Otherwise, denote $vk_{OT} = (vk_1, \ldots, vk_n)$ and compare if:

$$\begin{aligned} \hat{e}(g, S_1) &\stackrel{?}{=} \hat{e}(g_1, g_2) \cdot \hat{e}(S_2', u' \prod_{i=1}^{n} u_i^{vk_i}), \\ 1 &\stackrel{?}{=} \mathsf{Verify}_{OT}(vk_{OT}, S_3, m||S_1||S_2). \end{aligned}$$

Output 1 if the all of the above hold. Otherwise output 0.

– **UConvert**($\mathsf{param}, \mathsf{sk}$). The signer publishes his universal receipt $R = (\beta', \beta_1, \ldots, \beta_\ell)$.

– **UVerify**($\mathsf{param}, \mathsf{pk}, m, \sigma, R$). Upon input the signature $\sigma = (S_1, S_2, S_3, vk_{OT})$ on the message $m$ and the universal receipt $R = (\beta', \beta_1, \ldots, \beta_\ell)$, check if:

$$v' \stackrel{?}{=} g^{\beta'}, \qquad v_i \stackrel{?}{=} g^{\beta_i} \quad \text{for } 1 \leq i \leq \ell.$$

If they are not equal, output $\perp$. Otherwise compute $\bar{vk} = H(vk_{OT})$ and denote $vk_{OT} = (vk_1, \ldots, vk_n)$. Compare if:

$$\begin{aligned} \hat{e}(g, S_1) &\stackrel{?}{=} \hat{e}(g_1, g_2) \cdot \hat{e}(S_2^{1/(\beta' + \sum_{i=1}^{\ell} \beta_i \bar{vk}^i)}, u' \prod_{i=1}^{n} u_i^{vk_i}), \\ 1 &\stackrel{?}{=} \mathsf{Verify}_{OT}(vk_{OT}, S_3, m||S_1||S_2). \end{aligned}$$

Output 1 if all of the above hold. Otherwise output 0.

### 4.2 Security Result

**Theorem 1.** *The proposed convertible undeniable signature scheme is $(\epsilon, t, q_s)$-strongly unforgeable if the $(\epsilon', t')$-CDH assumption holds in $\mathbb{G}$, where*

$$\epsilon' \geq \frac{\epsilon}{2n+1}, \qquad t' = t + O\Big(q_s(\rho+\omega)\Big),$$

*and $\rho$, $\omega$ are the time for an exponentiation in $\mathbb{G}$ and for running $\mathsf{Kg}_{OT}$ and $\mathsf{Sign}_{OT}$ respectively.*

*Proof.* Assume there is a $(\epsilon, t, q_s)$-adversary $\mathcal{A}$. We are going to construct another PPT $\mathcal{B}$ that makes use of $\mathcal{A}$ to solve the CDH problem with probability at least $\epsilon'$ and in time at most $t'$.

$\mathcal{B}$ is given a CDH problem instance $(g, g^a, g^b)$. In order to use $\mathcal{A}$ to solve for the problem, $\mathcal{B}$ needs to simulates a challenger and the oracles for $\mathcal{A}$. $\mathcal{B}$ does it in the following way.

Setup. $\mathcal{B}$ runs $\mathsf{Kg}_{OT}(1^\lambda)$ for $2q_s$ times and obtains the pairs $(\mathsf{sk_t}, \mathsf{vk_t})$ for $1 \leq t \leq 2q_s$. $\mathcal{B}$ randomly selects the following integers:

- $x'_0 \in_R [0, 2n]$ ; $x'_1 \in_R [0, 2n]$ ; $y' \in_R \mathbb{Z}_p$, where $x'_0 \neq x'_1$.
- $x_i \in_R \{1, 2\}$, for $i = 1, \ldots, n$.
- $y_i \in_R \mathbb{Z}_p$, for $i = 1, \ldots, n$.

We further define the following functions for binary strings $\mathsf{vk_t} = (vk_{t,1}, \ldots, vk_{t,n})$ as follow:

$$F_0(\mathsf{vk_t}) = x'_0 + \sum_{i=1}^{n} x_i vk_{t,i}, \qquad F_1(\mathsf{vk_t}) = x'_1 + \sum_{i=1}^{n} x_i vk_{t,i}, \qquad J(\mathsf{vk_t}) = y' + \sum_{i=1}^{n} y_i vk_{t,i}.$$

For $j = 0, 1$, if there are at least $q_s$ number of $\mathsf{vk_t}$ such that $F_j(\mathsf{vk_t}) = 0$ for $\mathsf{vk_t} \in \{\mathsf{vk_1}, \ldots, \mathsf{vk_{2q_s}}\}$, then there must be at least $q_s$ number of $\mathsf{vk_t}$ satisfying $F_{1-j}(\mathsf{vk_t}) \neq 0$. Without loss of generality, assume $F_0(\mathsf{vk_t}) \neq 0$ holds for $\mathsf{t} = 1, \ldots, q_s$. We denote the function $F = F_0$ for simplicity.

$\mathcal{B}$ randomly picks $\beta', \beta_i \in \mathbb{Z}_p^*$ for $1 \leq i \leq \ell$ and sets $v' = g^{\beta'}$ and $v_i = g^{\beta_i}$. $\mathcal{B}$ constructs a set of public parameters as follow:

$$g, \qquad g_2 = g^b, \qquad u' = g_2^{x'_0} g^{y'}, \qquad u_i = g_2^{x_i} g^{y_i} \ \text{ for } \ 1 \leq i \leq n.$$

The signer's public key is $(g_1 = g^a, v', v_1, \ldots, v_\ell)$.

Denote $\bar{vk}_t = H(\mathsf{vk_t})$ and $G(\mathsf{vk_t}) = \beta' + \sum_{i=1}^{\ell} \beta_i \bar{vk}_t^i$. Note that we have the following equation:

$$u' \prod_{i=1}^{n} u_i^{vk_{t,i}} = g_2^{F(\mathsf{vk_t})} g^{J(\mathsf{vk_t})}, \qquad v' \prod_{i=1}^{\ell} v_i^{\bar{vk}_t^i} = g^{G(\mathsf{vk_t})}.$$

All the public parameters and the universal receipt $(\beta', \beta_1, \ldots, \beta_\ell)$ are passed to $\mathcal{A}$.

Oracles Simulation. $\mathcal{B}$ simulates the oracles as follow:

(*Signing oracle.*) Upon receiving the $t$-th signing oracle query for a message $m$, $\mathcal{B}$ retrieves the key pairs $(\mathsf{sk_t}, \mathsf{vk_t})$. $\mathcal{B}$ randomly chooses $r \in_R \mathbb{Z}_p$ and computes

$$S_1 = g_1^{-\frac{J(\mathsf{vk_t})}{F(\mathsf{vk_t})}} \big(g_2^{F(\mathsf{vk_t})} g^{J(\mathsf{vk_t})}\big)^r, \qquad S_2 = (g_1^{-\frac{1}{F(\mathsf{vk_t})}} g^{r_i})^{G(\mathsf{vk_t})}, \qquad S_3 = \mathsf{Sign}_{OT}(\mathsf{sk_t}, m||S_1||S_2).$$

By letting $\tilde{r} = r - \frac{a}{F(\mathsf{vk_t})}$, it can be verified that $(S_1, S_2, S_3, \mathsf{vk_t})$ is a signature, shown as follow:

$$\begin{aligned} S_1 &= g_1^{-\frac{J(\mathsf{vk_t})}{F(\mathsf{vk_t})}} (g_2^{F(\mathsf{vk_t})} g^{J(\mathsf{vk_t})})^r \\ &= g^{-\frac{aJ(\mathsf{vk_t})}{F(\mathsf{vk_t})}} (g_2^{F(\mathsf{vk_t})} g^{J(\mathsf{vk_t})})^{\frac{a}{F(\mathsf{vk_t})}} (g_2^{F(\mathsf{vk_t})} g^{J(\mathsf{vk_t})})^{-\frac{a}{F(\mathsf{vk_t})}} (g_2^{F(\mathsf{vk_t})} g^{J(\mathsf{vk_t})})^r \\ &= g^{-\frac{aJ(\mathsf{vk_t})}{F(\mathsf{vk_t})}} g_2^a g^{\frac{aJ(\mathsf{vk_t})}{F(\mathsf{vk_t})}} (g_2^{F(\mathsf{vk_t})} g^{J(\mathsf{vk_t})})^{\tilde{r}} \\ &= g_2^a (u' \prod_{j=1}^{n} u_j^{vk_{t,j}})^{\tilde{r}}, \end{aligned}$$

$$S_2 = (g_1^{-\frac{1}{F(\mathsf{vk_t})}} g^r)^{G(\mathsf{vk_t})} = (g^{r - \frac{a}{F(\mathsf{vk_t})}})^{G(\mathsf{vk_t})} = g^{G(\mathsf{vk_t})\tilde{r}} = (v' \prod_{i=1}^{\ell} v_i^{\bar{vk}_t^i})^{\tilde{r}}.$$

$\mathcal{B}$ outputs the signature $(S_1, S_2, S_3, \mathsf{vk_t})$. To the adversary, all signatures given by $\mathcal{B}$ are indistinguishable from the signatures generated by the signer. Notice that $F(\mathsf{vk_t}) \neq 0 \bmod p$ by the construction in the Setup phase.

Output. Finally $\mathcal{A}$ outputs a signature $\sigma^* = (S_1^*, S_2^*, S_3^*, vk_{OT}^*)$ for message $\mathsf{m}^*$. Denote $vk_{OT}^* = \{vk_1^*, \ldots, vk_n^*\}$. $\mathcal{B}$ checks if $F(vk_{OT}^*) = 0 \bmod p$. If not, $\mathcal{B}$ aborts. Otherwise $\mathcal{B}$ computes $\bar{vk}^* = H(vk_{OT}^*)$ and outputs

$$\frac{S_1^*}{{S_{2,1}^*}^{J(vk_{OT}^*)/G(vk_{OT}^*)}} = \frac{g_2^a\left(u' \prod_{i=1}^n u_i^{vk_i^*}\right)^r}{\left(v' \prod_{i=1}^{\ell} v_i^{\bar{vk}_1^{*i}}\right)^{rJ(vk_{OT}^*)/G(vk_{OT}^*)}} = \frac{g_2^a\left(g^{J(vk_{OT}^*)}\right)^r}{g^{rJ(vk_{OT}^*)}} = g^{ab},$$

which is the solution to the CDH problem instance.

Probability Analysis. For the simulation to complete without aborting, we require that in the challenge phase, $F(vk_{OT}^*) = 0 \bmod p$. We consider the following cases:

- If $vk_{OT}^* \in \{\mathsf{vk}_1, \ldots, \mathsf{vk}_{\mathsf{q_s}}\}$, and $\sigma^*$ is not the output from the signing oracle query, then $\mathcal{B}$ obtains a forgery of the one time signature $S_3^*$ with the message $m^*||S_1^*||S_2^*$.
- If $vk_{OT}^* \notin \{\mathsf{vk}_1, \ldots, \mathsf{vk}_{\mathsf{q_s}}\}$, observe that $\sum_{i=1}^n x_i vk_{t,i} \in [0, 2n]$, where $x_i \in \{1, 2\}$ and $vk_{t,i} \in \{0, 1\}$. Since $x_0'$ is chosen uniformly at random from $[0, 2n]$. Therefore

$$\Pr[F(vk_{OT}^*) = 0 \bmod p] = \frac{1}{2n+1}.$$

If the one time signature is secure, the probability of $\mathcal{B}$ not aborting is

$$\Pr[\mathsf{not\ abort}] \geq \frac{1}{2n+1}.$$

Time Complexity Analysis. The time complexity of $\mathcal{B}$ is determined as follows. There are $O(1)$ exponentiations of $\mathbb{G}$ element and one $\mathsf{Sign}_{OT}$ in the signing stage. There are $2q_s$ of $\mathsf{Kg}_{OT}$ in the setup stage. The time complexity of $\mathcal{B}$ is

$$t + O\Big(q_s(\rho + \omega)\Big).$$

□

**Theorem 2.** *The scheme is $(\epsilon, t, q_c, q_r, q_s)$-invisible if the $(\epsilon', t')$-decision linear assumption holds in $\mathbb{G}$, where*

$$\epsilon' \geq \frac{\epsilon}{2n+1}, \qquad t' = t + O\Big((q_s + q_r)\rho + q_c\tau + q_s\omega\Big),$$

*where $\rho$, $\tau$, $\omega$ are the time for an exponentiation in $\mathbb{G}$, for an exponentiation in $\mathbb{G}_T$ and for running $\mathsf{Kg}_{OT}$ and $\mathsf{Sign}_{OT}$ respectively, under the assumption that $\ell > q_s$.*

*Proof.* Assume there is a $(\epsilon, t, q_c, q_r, q_s)$-adversary $\mathcal{A}$. We are going to construct another PPT $\mathcal{B}$ that makes use of $\mathcal{A}$ to solve the decisional linear problem with probability at least $\epsilon'$ and in time at most $t'$.

$\mathcal{B}$ is given a decisional linear problem instance $(u, v, h, u^a, v^b, h^c)$. In order to use $\mathcal{A}$ to solve for the problem, $\mathcal{B}$ needs to simulates the oracles for $\mathcal{A}$. $\mathcal{B}$ does it in the following way.

Setup. $\mathcal{B}$ runs $\mathsf{Kg}_{OT}(1^\lambda)$ for $2q_s + 2$ times and obtains the pairs $(\mathsf{sk_t}, \mathsf{vk_t})$ for $1 \leq t \leq 2q_s + 2$. $\mathcal{B}$ randomly selects the following integers:

- $x_0' \in_R \mathbb{Z}_p$ ; $x_1' \in_R \mathbb{Z}_p$ ; $y' \in_R [0, 2n]$, where $x_0' \neq x_1'$.

- $x_i \in_R \mathbb{Z}_p$, for $i = 1, \ldots, n$.
- $y_i \in_R \{1, 2\}$, for $i = 1, \ldots, n$.

We further define the following functions for binary strings $\mathsf{vk_t} = (vk_{t,1}, \ldots, vk_{t,n})$ as follow:

$$F_0(\mathsf{vk_t}) = x'_0 + \sum_{i=1}^{n} x_i vk_{t,i}, \qquad F_1(\mathsf{vk_t}) = x'_1 + \sum_{i=1}^{n} x_i vk_{t,i}, \qquad J(\mathsf{vk_t}) = y' + \sum_{i=1}^{n} y_i vk_{t,i}.$$

For $j = 0, 1$, if there are at least $q_s + 1$ number of $\mathsf{vk_t}$ such that $F_j(\mathsf{vk_t}) = 0$ for $\mathsf{vk_t} \in \{\mathsf{vk_1}, \ldots, \mathsf{vk_{2q_s+2}}\}$, then there must be at least $q_s + 1$ number of $\mathsf{vk_t}$ satisfying $F_{1-j}(\mathsf{vk_t}) \neq 0$. Without loss of generality, assume $F_0(\mathsf{vk_t}) \neq 0$ holds for $\mathsf{t} = 1, \ldots, q_s + 1$. We denote the function $F = F_0$ for simplicity.

Assume that $\ell > q_s$. Denote the set $\bar{\mathcal{S}}$ as the set of numbers $\bar{vk_t} = H(\mathsf{vk_t})$, for $\mathsf{t} = 1, \ldots, q_s$. Also denote the set $\mathcal{S} = \mathbb{Z}_\ell \setminus \bar{S}$. We further define the following functions for any integer $\mathsf{vk_t} \in \mathbb{Z}_\ell$

$$G(\mathsf{vk_t}) = \prod_{i \in \mathcal{S}} (\bar{vk_t} - i) = \sum_{i=0}^{\ell - q_s} \gamma_i \bar{vk_t}^{\,i} \qquad \text{and} \qquad K(\mathsf{vk_t}) = \prod_{i \in \bar{\mathcal{S}}} (\bar{vk_t} - i) = \sum_{i=0}^{q_s} \alpha_i \bar{vk_t}^{\,i},$$

for some $\gamma_i, \alpha_i \in \mathbb{Z}_p$. For consistency, define $\gamma_{\ell - q_s + 1} = \ldots = \gamma_\ell = \alpha_{q_s + 1} = \ldots = \alpha_\ell = 0$.

$\mathcal{B}$ constructs a set of public parameters as follow:

$$g = u, \qquad g_2 = h, \qquad u' = g_2^{x'} g^{y'}, \qquad u_i = g_2^{x_i} g^{y_i} \;\text{ for }\; 1 \leq i \leq n.$$

The signer's public key is:

$$g_1 = u^a, \qquad v' = v^{\alpha_0} g^{\gamma_0}, \qquad v_i = v^{\alpha_i} g^{\gamma_i} \;\text{ for }\; 1 \leq i \leq \ell.$$

Note that we have the following equation:

$$u' \prod_{i=1}^{n} u_i^{vk_{t,i}} = g_2^{F(\mathsf{vk_t})} g^{J(\mathsf{vk_t})}, \qquad v' \prod_{i=1}^{\ell} v_i^{\bar{vk_t}^{\,i}} = g^{G(\mathsf{vk_t})} v^{K(\mathsf{vk_t})},$$

where $\bar{vk_t} = H(\mathsf{vk_t})$. All public parameters are passed to $\mathcal{A}$. $\mathcal{B}$ also maintains an empty list $\mathcal{L}$.

Oracles Simulation. $\mathcal{B}$ simulates the oracles as follow:

(*Signing oracle.*) Upon receiving the $t$-th signing oracle query for a message $m$, $\mathcal{B}$ retrieves the key pairs $(\mathsf{sk_t}, \mathsf{vk_t})$. Note that by the construction in setup, we have $F(\mathsf{vk_t}) \neq 0 \bmod p$ and $K(\mathsf{vk_t}) = 0 \bmod p$. $\mathcal{B}$ randomly chooses $r \in_R \mathbb{Z}_p$ and computes

$$S_1 = g_1^{-\frac{J(\mathsf{vk_t})}{F(\mathsf{vk_t})}} \left(g_2^{F(\mathsf{vk_t})} g^{J(\mathsf{vk_t})}\right)^{r_i}, \qquad S_2 = (g_1^{-\frac{1}{F(\mathsf{vk_t})}} g^{r_i})^{G(\mathsf{vk_t})}, \qquad S_3 = \mathsf{Sign}_{OT}(\mathsf{sk_t}, m||S_1||S_2).$$

Same as the above proof, the signature $\sigma = (S_1, S_2, S_3, \mathsf{vk_t})$ is valid. $\mathcal{B}$ puts $(m, \sigma)$ into the list $\mathcal{L}$ and then outputs the signature $\sigma$. To the adversary, all signatures given by $\mathcal{B}$ are indistinguishable from the signatures generated by the signer.

(*Confirmation/Disavowal oracle.*) Upon receiving a signature $\sigma = (S_1, S_2, S_3, \mathsf{vk_t})$ for message $m$, $\mathcal{B}$ checks whether $(m, \sigma)$ is in $\mathcal{L}$. If so, $\mathcal{B}$ outputs $\mathsf{Valid}$ and runs the confirmation protocol with $\mathcal{A}$, to show that $(L, M, N, O)$ in equation (1) are DH tuples. Notice that since $\mathcal{B}$ knows discrete logarithm of $N$ with base $L$ ( $= 1/G(\mathsf{vk_t})$), it can simulate the interactive proof perfectly. Note that $G(\mathsf{vk_t}) \neq 0$ if $(m, \sigma) \in \mathcal{L}$.

If the signature is not in $\mathcal{L}$, $\mathcal{B}$ outputs $\mathsf{Invalid}$ and runs the disavowal protocol with $\mathcal{A}$. By theorem 1, the signature is strongly unforgeable if the CDH assumption holds. $\mathcal{B}$ runs the oracle incorrectly only if $\mathcal{A}$ can forge a signature. However if one can solve the CDH problem, he can also solve the decision linear problem.

(*Receipt generating oracle.*) Upon receive a signature $\sigma = (S_1, S_2, S_3, \mathsf{vk_t})$ for message $m$, $\mathcal{B}$ checks whether $(m, \sigma)$ is in $\mathcal{L}$. If so, $\mathcal{B}$ outputs $S_2' = S_{2,1}^{1/G(\mathsf{vk_t})}$, which is a valid individual receipt for the signature. Otherwise, $\mathcal{B}$ returns $\perp$ which indicates that $\sigma$ is not a valid signature.

Challenge. $\mathcal{A}$ gives $m^*$ to $\mathcal{B}$ as the challenge message. $\mathcal{B}$ retrieves the key pairs $(\mathsf{sk_{q_s+1}}, \mathsf{vk_{q_s+1}})$. Denote $\mathsf{vk_{q_s+1}} = \{vk_1^*, \dots, vk_n^*\}$ and $v\bar{k}^* = H(\mathsf{vk_{q_s+1}})$. Note by the construction in setup, we have $F(\mathsf{vk_{q_s+1}}) \neq 0 \bmod p$. We can also see that if $G(\mathsf{vk_{q_s+1}}) \neq 0 \bmod p$, then $\mathsf{vk_{q_s+1}} \in \bar{\mathcal{S}}$. It implies that $H(\mathsf{vk_{q_s+1}}) = H(\mathsf{vk_t})$ for some $t \in [1, \dots, q_s]$. If the hash function $H$ is collision resistant, then $G(\mathsf{vk_{q_s+1}}) = 0 \bmod p$.

If $J(\mathsf{vk_{q_s+1}}) \neq 0 \bmod p$, $\mathcal{B}$ aborts. Otherwise, $\mathcal{B}$ computes:

$$S_1^* = h^c, \qquad S_2^* = v^{bK(\mathsf{vk_{q_s+1}})/F(\mathsf{vk_{q_s+1}})}, \qquad S_3^* = \mathsf{Sign}_{OT}(\mathsf{sk_{q_s+1}}, m^*||S_1^*||S_2^*).$$

and returns $(S_1^*, S_2^*, S_3^*, \mathsf{vk_{q_s+1}})$ to $\mathcal{A}$.

Output. Finally $\mathcal{A}$ outputs a bit $b'$. $\mathcal{B}$ returns $b'$ as the solution to the decision linear problem. Notice that if $c = a + b$, then:

$$S_1^* = g_2^{a+b} = g_2^a (g_2^{F(\mathsf{vk_{q_s+1}})})^{b/F(\mathsf{vk_{q_s+1}})} = g_2^a (u' \prod_{i=1}^{n} u_i^{m_i^*})^{b/F(\mathsf{vk_{q_s+1}})},$$

$$S_2^* = v^{bK(\mathsf{vk_{q_s+1}})/F(\mathsf{vk_{q_s+1}})} = (v' \prod_{i=1}^{\ell} v_i^{v\bar{k}^*{}_i})^{b/F(\mathsf{vk_{q_s+1}})}.$$

Probability Analysis. For the simulation to complete without aborting, we require that in the challenge phase, $J(\mathsf{vk_{q_s+1}}) = 0 \bmod p$. Observe that $\sum_{i=1}^{n} y_i vk_{t,i} \in [0, 2n]$, where $y_i \in \{1, 2\}$ and $vk_{t,i} \in \{0, 1\}$. Since $y'$ is chosen uniformly at random from $[0, 2n]$. Therefore

$$\Pr[J(\mathsf{vk_{q_s+1}}) = 0 \bmod p] = \frac{1}{2n+1}.$$

The probability of $\mathcal{B}$ not aborting is

$$\Pr[\mathsf{not\ abort}] \geq \frac{1}{2n+1}.$$

Time Complexity Analysis. The time complexity of $\mathcal{B}$ is determined as follows. There are $O(1)$ exponentiations of $\mathbb{G}$ element and one $\mathsf{Sign}_{OT}$ in the signing stage. There are $O(1)$ exponentiations of $\mathbb{G}_T$ element in the confirm/disavow stage. There are $O(1)$ exponentiations of $\mathbb{G}$ element in the receipt generating stage. There are $2q_s + 2$ of $\mathsf{Kg}_{OT}$ in the setup stage. The time complexity of $\mathcal{B}$ is

$$t + O\Big((q_s + q_r)\rho + q_c\tau + q_s\omega\Big).$$

□

**Theorem 3.** *The scheme is $(\epsilon, t, q_c, q_s)$-secure against impersonation if the $(\epsilon'$, $t')$-discrete logarithm assumption holds in $\mathbb{G}$, where*

$$\epsilon' \geq \frac{1}{2}(\epsilon - \frac{1}{p})^2, \qquad t' = t + O\Big(q_s\rho + q_c\tau + q_s\omega\Big),$$

*where $\rho$, $\tau$, $\omega$ are the time for an exponentiation in $\mathbb{G}$, for an exponentiation in $\mathbb{G}_T$ and for running $\mathsf{Kg}_{OT}$ and $\mathsf{Sign}_{OT}$ respectively.*

*Proof.* Assume there is a $(\epsilon, t, q_c, q_s)$-adversary $\mathcal{A}$. We are going to construct another PPT $\mathcal{B}$ that makes use of $\mathcal{A}$ to solve the discrete logarithm problem with probability at least $\epsilon'$ and in time at most $t'$. $\mathcal{B}$ is given a discrete logarithm problem instance $(g, g^a)$. The remaining proof is very similar to the proof of theorem 1, so we sketch the proof here.

With 1/2 probability, $\mathcal{B}$ sets $g_1 = g^a$ and hence the user secret key is $a$. The oracle simulation is the same as the proof in theorem 1, except that $\mathcal{B}$ now knows $b = \log_g g_2$. At the end of the game, $\mathcal{A}$ outputs a message-signature pair $(m^*, \sigma^*)$ and a bit $b^*$. For either $b^* = 0/1$, $\mathcal{B}$ can extract $a$ with probability 1/2, using the extractor of the proof of knowledge protocol.

With 1/2 probability, $\mathcal{B}$ sets $v' = g^a$ and hence $\mathcal{B}$ knows the signing key $\alpha$. $\mathcal{B}$ can simulate the oracles perfectly with $\alpha$. At the end of the game, $\mathcal{A}$ outputs a message-signature pair $(m^*, \sigma^*)$ and a bit $b^*$. For either $b^* = 0/1$, $\mathcal{B}$ can extract $a + \sum_{i=1}^{\ell} \beta_i v \bar{k^*}^i$ with probability 1/2, using the extractor of the proof of knowledge protocol. Hence $\mathcal{B}$ can find $a$.

Probability Analysis. For the simulation to complete without aborting, we require that $\mathcal{B}$ correctly extract $a$ at the end of the game. By Reset Lemma, it happens with probability at least $\frac{1}{2}(\epsilon - \frac{1}{p})^2$. We have

$$\epsilon' \geq \frac{1}{2}(\epsilon - \frac{1}{p})^2.$$

Time Complexity Analysis. The time complexity of $\mathcal{B}$ is determined as follows. There are $O(1)$ exponentiations of $\mathbb{G}$ element and one $\mathsf{Sign}_{OT}$ in the signing stage. There are $O(1)$ exponentiations of $\mathbb{G}_T$ element and $O(1)$ modular addition in $\mathbb{Z}_p$ in the confirm/disavow stage. There are $2q_s + 2$ of $\mathsf{Kg}_{OT}$ in the setup stage. The time complexity of $\mathcal{B}$ is

$$t + O\Big(q_s\rho + q_c\tau + q_s\omega\Big).$$

□

## 5 Conclusion

In this paper, we propose the first convertible undeniable signatures without random oracles in pairings. Comparing with the part of undeniable signatures, our scheme is better than the existing undeniable signatures without random oracles [26] by using more standard assumption in the security proofs.

We improve the earlier version of our scheme in [38] in several ways. Firstly, our current scheme provides strong unforgeability while the earlier version provides existential unforgeability. Secondly, our current scheme fixes a flaw in the proof of invisibility [34]. Finally, our current scheme significantly reduces the reduction loss in the security proof. The earlier version of our scheme [38] has an exponential reduction loss. Our current scheme has $O(n)$ reduction loss only.

In 2009, Phong *et al.* [34] proposed another convertible undeniable signatures without random oracles in pairings. We consider their concrete scheme $\mathsf{SCUS}_2$ for comparison purpose. The $\mathsf{SCUS}_2$ scheme is more efficient than our current scheme, since it has less public keys and less multiplication in the $\mathsf{USign}$ algorithm. However, our current scheme uses the weaker CDH assumption for unforgeability, while the $\mathsf{SCUS}_2$ scheme uses the $q$-SDH assumption.

## References


1. G. Ateniese, J. Camenisch, S. Hohenberger, and B. de Medeiros. Practical group signatures without random oracles. Cryptology ePrint Archive, Report 2005/385, 2005. http://eprint.iacr.org/.
2. M. Bellare, A. Boldyreva, and A. Palacio. An uninstantiable random-oracle-model scheme for a hybrid-encryption problem. In *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 171–188. Springer, 2004.

3. M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *CCS '93*, pages 62–73. ACM Press, 1993.
4. M. Bellare and S. Shoup. Two-tier signatures, strongly unforgeable signatures, and fiat-shamir without random oracles. In *PKC 2007*, volume 4450 of *LNCS*, pages 201–216. Springer, 2007.
5. A. Bender, J. Katz, and R. Morselli. Ring signatures: Stronger definitions, and constructions without random oracles. In *TCC 2006*, volume 3816 of *LNCS*, pages 60–79. Springer, 2006.
6. D. Boneh, X. Boyen, and H. Shacham. Short group signatures. In *CRYPTO 2004*, volume 3152 of *LNCS*, pages 41–55. Springer, 2004.
7. J. Boyar, D. Chaum, I. Damgård, and T. P. Pedersen. Convertible undeniable signatures. In *CRYPTO '90*, volume 537 of *LNCS*, pages 189–205. Springer, 1991.
8. X. Boyen and B. Waters. Anonymous hierarchical identity-based encryption (without random oracles). In *CRYPTO 2006*, volume 4117 of *LNCS*, pages 290–307. Springer, 2006.
9. X. Boyen and B. Waters. Compact group signatures without random oracles. In *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 427–444. Springer, 2006.
10. R. Canetti, O. Goldreich, and S. Halevi. The random oracle methodology, revisited. In 13*th ACM Symp. on Theory of Computing*, pages 209–128. ACM Press, 1998.
11. D. Chaum. Designated confirmer signatures. In *EUROCRYPT '94*, volume 950 of *LNCS*, pages 86–91. Springer, 1994.
12. D. Chaum and H. van Antwerpen. Undeniable signatures. In *CRYPTO '89*, volume 435 of *LNCS*, pages 212–216. Springer, 1989.
13. S. S. Chow, J. K. Liu, V. K. Wei, and T. H. Yuen. Ring signatures without random oracles. In *ASIACCS 2006*, pages 297–302. ACM Press, 2006.
14. I. Damgård and T. P. Pedersen. New convertible undeniable signature schemes. In *EUROCRYPT '96*, volume 1070 of *LNCS*, pages 372–386. Springer, 1996.
15. R. Gennaro, T. Rabin, and H. Krawczyk. Rsa-based undeniable signatures. *J. Cryptology*, 13(4):397–416, 2000.
16. J. Groth and A. Sahai. Efficient non-interactive proof systems for bilinear groups. In *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 415–432. Springer, 2008.
17. F. Guo, Y. Mu, and W. Susilo. How to prove security of a signature with a tighter security reduction. In *ProvSec 2009*, volume 5848 of *LNCS*, pages 90–103. Springer, 2009.
18. S. Hohenberger, G. Rothblum, A. Shelat, and V. Vaikuntanathan. Securely obfuscating re-encryption. In *TCC 2007*, volume 4392 of *LNCS*, pages 233–252. Springer, 2007.
19. X. Huang, Y. Mu, W. Susilo, and W. Wu. A generic construction for universally-convertible undeniable signatures. In *CANS 2007*, volume 4856 of *LNCS*, pages 15–33. Springer, 2007.
20. X. Huang, Y. Mu, W. Susilo, and W. Wu. Provably secure pairing-based convertible undeniable signature with short signature length. In *Pairing 2007*, volume 4575 of *LNCS*, pages 367–391. Springer, 2007.
21. M. Jakobsson, K. Sako, and R. Impagliazzo. Designated verifier proofs and their applications. In *EUROCRYPT '96*, volume 1070 of *LNCS*, pages 143–154. Springer, 1996.
22. A. Kiayias and H.-S. Zhou. Concurrent blind signatures without random oracles. In *SCN 2006*, volume 4116 of *LNCS*, pages 49–62. Springer, 2006.
23. K. Kurosawa and J. Furukawa. Universally composable undeniable signature. In *ICALP 2008*, volume 5126 of *LNCS*, pages 524–535. Springer, 2008.
24. K. Kurosawa and S.-H. Heng. 3-move undeniable signature scheme. In *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 181–197. Springer, 2005.
25. K. Kurosawa and T. Takagi. New approach for selectively convertible undeniable signature schemes. In *ASIACRYPT 2006*, volume 4284 of *LNCS*, pages 428–443, 2006.
26. F. Laguillaumie and D. Vergnaud. Short undeniable signatures without random oracles: The missing link. In *INDOCRYPT 2005*, volume 3797 of *LNCS*, pages 283–296. Springer, 2005.
27. F. Laguillaumie and D. Vergnaud. Time-selective convertible undeniable signatures. In *CT-RSA 2005*, volume 3376 of *LNCS*, pages 154–171. Springer, 2005.
28. B. Libert and J.-J. Quisquater. Identity based undeniable signatures. In *CT-RSA 2004*, volume 2964 of *LNCS*, pages 112–125. Springer, 2004.
29. M. Michels, H. Petersen, and P. Horster. Breaking and repairing a convertible undeniable signature scheme. In *CCS '96*, pages 148–152. ACM Press, 1996.
30. M. Michels and M. Stadler. Efficient convertible undeniable signature schemes. In *SAC '97*, pages 231–244, 1997.
31. J. Monnerat and S. Vaudenay. Generic homomorphic undeniable signatures. In *ASIACRYPT 2004*, volume 3329 of *LNCS*, pages 354–371. Springer, 2004.

32. W. Ogata, K. Kurosawa, and S.-H. Heng. The security of the fdh variant of chaum's undeniable signature scheme. *IEEE Transactions on Information Theory*, 52(5):2006–2017, 2006.
33. T. Okamoto. Designated confirmer signatures and public key encryption are equivalent. In *CRYPTO '94*, volume 939 of *LNCS*, pages 61–74. Springer, 1994.
34. L. T. Phong, K. Kurosawa, and W. Ogata. New dlog-based convertible undeniable signature schemes in the standard model. Cryptology ePrint Archive, Report 2009/394, 2009. http://eprint.iacr.org/.
35. L. T. Phong, K. Kurosawa, and W. Ogata. New rsa-based (selectively) convertible undeniable signature schemes. In *AFRICACRYPT 2009*, volume 5580 of *LNCS*, pages 116–134. Springer, 2009.
36. H. Wang, Y. Zhang, and D. Feng. Short threshold signature schemes without random oracles. In *INDOCRYPT 2005*, volume 3797 of *LNCS*, pages 297–310. Springer, 2005.
37. B. Waters. Efficient identity-based encryption without random oracles. In *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 114–127. Springer, 2005.
38. T. H. Yuen, M. H. Au, J. K. Liu, and W. Susilo. (convertible) undeniable signatures without random oracles. In *ICICS 2007*, volume 4861 of *LNCS*, pages 83–97. Springer, 2007.
39. R. Zhang, J. Furukawa, and H. Imai. Short signature and universal designated verifier signature without random oracles. In *ACNS 2005*, volume 3531 of *LNCS*, pages 483–498. Springer, 2005.