# Toward a Generic Construction of Universally Convertible Undeniable Signatures from Pairing-Based Signatures

Laila El Aimani

b-it (Bonn-Aachen International Center for Information Technology),
Dahlmannstr. 2, D-53113 Bonn, Germany
elaimani@bit.uni-bonn.de

**Abstract.** Undeniable signatures were proposed to limit the verification property of ordinary digital signatures. In fact, the verification of such signatures cannot be attained without the help of the signer, via the confirmation/denial protocols. Later, the concept was refined to give the possibility of converting the issued undeniable signatures into ordinary ones by publishing a *universal* receipt that turns them publicly verifiable.

In this paper, we present the first generic construction for universally convertible undeniable signatures from certain weakly secure cryptosystems and any secure digital signature scheme. Next, we give two specific approaches for building universally convertible undeniable signatures from a large class of pairing-based signatures. These methods find a nice and practical instantiation with known encryption and signature schemes. For instance, we achieve the most efficient undeniable signatures with regard to the signature length and cost, the underlying assumption and the security model. We believe these constructions could be an interesting starting point to develop more efficient schemes or give better security analyses of the existing ones.

**Keywords:** Undeniable signatures, Pairing-based signatures, Generic construction.

## 1 Introduction

Undeniable signatures were originally introduced in 1990 by Chaum and van Antwerpen [8] to limit the self-authenticating property of digital signatures. In fact, the verification algorithm in these signatures is replaced by a confirmation (denial) protocol between the verifier and the signer, in which the verifier learns the validity (invalidity) of the issued signature without being able to transfer his conviction to a third person. This cryptographic primitive proved valuable in many applications where privacy is a big concern, e.g., licensing software.

In 1991, the notion of undeniable signature was boosted by Boyar et al. [3] to allow the conversion of a selected undeniable signature into an ordinary one by releasing a piece of information at a later time. The model supported also the universal conversion achieved by publishing a universal receipt (by the signer) that transforms all undeniable signatures into publicly verifiable ones.

D.R. Chowdhury, V. Rijmen, and A. Das (Eds.): INDOCRYPT 2008, LNCS 5365, pp. 145–157, 2008.
© Springer-Verlag Berlin Heidelberg 2008

### 1.1 Related Work

Since the introduction of undeniable signatures, a series of proposals sprang up, covering a variety of different aspects. Pairing-based signatures [1] have received a lot of attention in these settings. Actually, most such signatures include in the verification equation a pairing computation between a part of the signature and some other parameters. Therefore, if we implement the same signature in a non bilinear group, namely a group where the Decisional Diffie-Hellman problem (DDH) is intractable, the resulting signature cannot be publicly verifiable. Hence, the signer must perform a proof of equality/inequality of two discrete logarithms with the verifier. Such a duality between pairing-based signatures and undeniable signatures has been illustrated in the literature by some proposals, e.g., the BLS signatures [2] whose undeniable variant are the early Chaum and van Antwerpen [8] signatures or Boneh and Boyen's signatures [1] which resulted in Laguillaumie and Vergnaud's undeniable signatures [12]. All these signatures inherit the security properties of their underlying digital signatures and have their invisibility based on a variant of the DDH problem.

Unfortunately, this approach does not give the possibility of converting the resulting signatures. A tantalizing challenge is to propose a general approach that constructs undeniable signatures from (a large category of) pairing-based signatures with the possibility of converting them to ordinary ones.

### 1.2 Our Contributions

We propose the first generic construction of universally convertible undeniable signatures from secure digital signatures and some weakly secure cryptosystems. Our design uses the "encryption of a signature" method [2] and relaxes the security requirement on the underlying cryptosystem, without compromising the overall security. As a consequence, we allow malleable cryptosystems in our design which impacts positively the efficiency of the confirmation/denial protocols.

Next, we give an efficient generic construction of universally convertible undeniable signatures. In fact, following the same principle, we shrink the set of signatures, upon which we build the undeniable signatures, down to a certain class of pairing-based signatures and we use an appropriate Key Encapsulation Mechanism. This construction finds a very efficient instantiation and results in the most efficient universally convertible undeniable signature scheme without random oracles and whose security rests on standard assumptions.

Finally, we enlarge the set of pairing-based signatures to include most proposals that appeared in the literature so far. In this way, the resulting undeniable signatures inherit the same virtues of the underlying digital signatures and acquire other interesting properties concerning their invisibility.

[1] See Section 2 for definitions of pairings, bilinear groups, etc...

[2] This method has been successfully used in a number of primitives such as designated confirmer signatures [5]. It consists in generating a signature on the message to be signed, then encrypting it. The validity or invalidity of the resulting signature are checked via concurrent proofs of knowledge.

## 2 Preliminaries

### 2.1 Bilinear Maps

**Definition 1.** *Let* $(\mathbb{G}, +)$ *and* $(\mathbb{H}, \times)$ [3] *be groups of prime order* $d$*. Let* $P$ *be a generator of* $\mathbb{G}$*.* $\mathbb{G}$ *is called a* bilinear group *if there exists a map* $e : \mathbb{G} \times \mathbb{G} \to \mathbb{H}$*, with the following properties:*

1. *bilinearity: for all* $(P, Q) \in \mathbb{G}^2$ *and* $a, b \in \mathbb{Z}_d$*,* $e(aP, bQ) = e(P, Q)^{ab}$*,*
2. *efficient computability for any input pair, and*
3. *non-degeneracy:* $e(P, P) \neq 1_{\mathbb{H}}$*.*

### 2.2 Digital Signatures

A signature scheme $\Sigma$ comprises three algorithms, keygen, sign, and verify:

- keygen is a probabilistic key generation algorithm which returns pairs of private and public keys $(\mathsf{sk}, \mathsf{pk})$ depending on the security parameter $k$,
- sign is a signing algorithm which takes on input a private key sk and a plaintext $m$ and returns a signature $\sigma$, and
- verify is a deterministic algorithm which takes on input a public key pk, a signature $\sigma$ and outputs 1 if the signature is valid and 0 otherwise.

**Definition 2.** *A signature scheme is said to be* $(t, \epsilon, q_s)$*-EUF-CMA secure if no adversary* $\mathcal{A}$*, operating in time* $t$ *and issuing at most* $q_s$ *queries, wins the following game with probability greater than* $\epsilon$*, where the probability is taken over all the random choices:*
***Setup.*** *$\mathcal{A}$ is given the public parameters of the given signature scheme.*
***Queries.*** *$\mathcal{A}$ queries the challenger for signatures on at most $q_s$ messages.*
***Output.*** *$\mathcal{A}$ outputs a pair $(m, \sigma)$ and wins the game if $m$ has not been queried before and* $\mathsf{verify}_{\mathsf{pk}}(m, \sigma) = 1$*.*

### 2.3 Public-Key Encryption Schemes

An asymmetric encryption scheme comprises the following algorithms:

- keygen is a probabilistic key generation algorithm which returns pairs of private and public keys $(\mathsf{sk}, \mathsf{pk})$ depending on the security parameter $k$,
- encrypt is a probabilistic encryption algorithm which takes on input a public key pk and a plaintext $m$, and returns a ciphertext $c$, and
- decrypt is a deterministic decryption algorithm which takes on input a secret key sk and a ciphertext $c$, and returns the corresponding plaintext $m$ or $\perp$.

A cryptosystem provides indistinguishability (IND) if it is difficult to distinguish pairs of ciphertexts based on the messages they encrypt. In case the adversary against the scheme has access to a decryption oracle, the scheme is said to be indistinguishable under chosen ciphertext attacks (IND-CCA), otherwise it is indistinguishable under chosen plaintext attacks (IND-CPA). Formal definitions can be found in [4].

[3] In the rest of the document, the group $\mathbb{G}$ is denoted additively whereas the group $\mathbb{H}$ is denoted multiplicatively.

### 2.4 Key Encapsulation Mechanisms (KEM)

A KEM is a tuple of algorithms $\mathcal{K} = (\mathsf{keygen}, \mathsf{encap}, \mathsf{decap})$ where

- $\mathsf{keygen}$ probabilistically generates a key pair $(\mathsf{sk}, \mathsf{pk})$,
- $\mathsf{encap}$, or the *encapsulation* algorithm which, on input a random nonce $r$ and the public key $\mathsf{pk}$, generates a *session key* denoted $k$ and its *encapsulation* $c$, and
- $\mathsf{decap}$, or the *decapsulation* algorithm. Given the private key $\mathsf{sk}$ and the element $c$, this algorithm computes the decapsulation $k$ of $c$, or returns $\perp$ if $c$ is invalid.

**Definition 3.** *A KEM is said to be $(t, \epsilon)$-IND-CPA secure if no adversary $\mathcal{A}$, operating in time $t$, wins the following game with probability greater than $\epsilon$:*

- ***Phase 1.*** *$\mathcal{A}$ gets the parameters of the KEM from his challenger.*
- ***Challenge.*** *The challenger computes a given encapsulation $c^\star$, then picks uniformly at random a bit $b$ from $\{0, 1\}$. If $b = 1$, then he sets $k^\star$ to $k_1$ where $k_1 = \mathsf{decap}(c^\star)$. Otherwise, he sets $k^\star$ to a uniformly chosen string from the session keys space. The challenge is $(c^\star, k^\star)$.*
- ***Phase 2.*** *$\mathcal{A}$ outputs a bit $b'$ (representing his guess of $k^\star$ being the decapsulation of $c^\star$) and wins the game if $b = b'$. We define $\mathcal{A}$'s advantage as $\mathsf{Adv}(\mathcal{A}) = |\Pr[b = b'] - \frac{1}{2}|$, where the probability is taken over the random choices of the adversary $\mathcal{A}$ and the challenger.*

**The Hybrid Encryption Paradigm.** It consists in combining KEMs with secure secret key encryption algorithms or Data Encapsulation Mechanisms (DEMs) to build encryption schemes. In fact, one can fix a session key $k$ using the KEM, then uses it to encrypt a message using an efficient DEM. Decryption is achieved by first recovering the key from the encapsulation (part of the ciphertext) then applying the DEM decryption algorithm. It can be shown that one can obtain an IND-CPA cryptosystem from an IND-CPA KEM combined with a DEM indistinguishable under a one time attack (IND-OT). We refer to [11] for the necessary and sufficient conditions on KEMs and DEMs in order to obtain a certain level of security for the resulting hybrid encryption scheme.

## 3 Universally Convertible Undeniable Signatures (UCUS)

### 3.1 Definition

**Setup.** On input the security parameter $k$, outputs the public parameters.
**Key Generation.** Generates probabilistically a key pair $(\mathsf{sk}, \mathsf{pk})$.
**Signature.** On input the public parameters, the private key $\mathsf{sk}$ and a message $m$, outputs an undeniable signature $\mu$.
**Verification.** This is an algorithm run by the signer to check the validity of an undeniable signature $\mu$ issued on $m$, using his private key $\mathsf{sk}$.

**Confirmation/Denial Protocol.** These are interactive protocols between a prover and a verifier. Their common input consists of the public parameters of the scheme, the signature $\mu$ and the message $m$ in question. The prover, that is the signer, uses his private key $\mathsf{sk}$ to convince the verifier of the validity (invalidity) of the signature $\mu$ on $m$.

**Universal Conversion.** Releases a universal receipt, using $\mathsf{sk}$, that makes all undeniable signatures universally verifiable.

**Universal Verification.** On input a signature, a message, a receipt and the public key $\mathsf{pk}$, outputs 1 if the signature is valid and 0 otherwise.

### 3.2 Security Model

In addition to the completeness, soundness and non-transferability of the proofs inherent to the confirmation/denial protocols, a convertible undeniable signature scheme requires two further properties, that are unforgeability and invisibility.

**Unforgeability.** The natural security requirement that a universally convertible signature scheme should fulfill is the existential unforgeability against a chosen message attack (EUF-CMA). It is defined through the following game.

- **Setup.** The adversary $\mathcal{A}$ is given the public parameters of the scheme in addition to the universal receipt.
- **Queries.** $\mathcal{A}$ queries the signing oracle adaptively on at most $q_s$ messages. Note that there will be no need to query the confirmation/denial oracles since $\mathcal{A}$ has the universal receipt at his disposal.
- **Output.** At the end, $\mathcal{A}$ outputs a pair consisting of a message $m$, that has not been queried before, and a string $\mu$. $\mathcal{A}$ wins the game if $\mu$ is a valid undeniable signature on $m$.

We say that a universally convertible undeniable signature scheme is $(t, \epsilon, q_s)$-EUF-CMA secure if there is no adversary, operating in time $t$, that wins the above game with probability greater than $\epsilon$.

**Invisibility.** Invisibility against a chosen message attack (INV-CMA) is defined through the following game between an attacker $\mathcal{A}$ and his challenger $\mathcal{R}$.

- $\mathcal{A}$ gets the parameters of the scheme from $\mathcal{R}$.
- **Phase 1.** $\mathcal{A}$ adaptively query the signing and confirmation/denial oracles.
- **Challenge.** Eventually, $\mathcal{A}$ outputs a message $m^\star$ that has not been queried before to the signing oracle and requests a challenge signature $\mu^\star$. $\mathcal{R}$ picks a bit $b \in_R \{0, 1\}$. If $b = 1$, then $\mu^\star$ is generated as usual using the signing oracle, otherwise it is chosen uniformly at random from the signatures space.
- **Phase 2.** $\mathcal{A}$ can adaptively query the previous oracles with the exception of not querying $m^\star$ to the signing oracle or $(m^\star, \mu^\star)$ to the verification oracles.
- **Output.** $\mathcal{A}$ outputs a bit $b'$ representing his guess on $\mu^\star$ being a valid signature on $m^\star$. He wins the game if $b = b'$. We define $\mathcal{A}$'s advantage as $\mathsf{Adv}(\mathcal{A}) = |\Pr[b = b'] - \frac{1}{2}|$.

We say that a convertible undeniable signature scheme is $(t, \epsilon, q_s, q_v)$-INV-CMA secure if no adversary operating in time $t$, issuing $q_s$ queries to the signing oracle and $q_v$ queries to the confirmation/denial oracles wins the above game with advantage greater than $\epsilon$.

## 4 A Systematic Approach for UCUS from Some Cryptosystems and Digital Signatures

### 4.1 Design Principle

We use the "encryption of a signature" method. Thus, we first generate a digital signature on the message to be signed, then encrypt the resulting signature using a suitable cryptosystem obtained from the hybrid encryption paradigm. Confirmation or denial of the resulting signatures exist by virtue of Goldreich et al.'s result [10]. In fact, the verification and decryption algorithms in a signature scheme and a cryptosystem respectively define an NP (co-NP) language for which there exists a zero knowledge proof system.

This method has been in use for some time ago. For instance, Camenisch and Michels [5] used it for designated confirmer signatures. One of the main differences between the two proposals dwells in the security assumption on the cryptosystem. We actually require only IND-CPA secure KEMs (thus IND-CPA cryptosystems), as we do not allow individual conversions of the undeniable signatures, versus IND-CCA cryptosystems. The consequences of this are twofold. First, we require a weak security notion on the cryptosystem without compromising the overall security. This gives many and simpler choices for the cryptosystem to be used. Second, we allow malleable cryptosystems in our construction, which impacts positively the confirmation/denial protocols efficiency. In fact, cryptosystems with homomorphic properties possess efficient decryption proofs of knowledge, i.e, one can prove efficiently the knowledge of the plaintext corresponding to a given ciphertext. Such schemes are not ruled out from our design.

### 4.2 Proposed Construction

Let $\Sigma$ be a digital signature scheme given by $\Sigma.\mathsf{keygen}$ which generates a key pair (private key = $\Sigma.\mathsf{sk}$, public key= $\Sigma.\mathsf{pk}$), $\Sigma.\mathsf{sign}$ and $\Sigma.\mathsf{verify}$.

Let furthermore $\Gamma$ be a cryptosystem obtained using the hybrid encryption paradigm and described by $\Gamma.\mathsf{keygen}$ (that generates the pair (private key = $\Gamma.\mathsf{sk}$, public key= $\Gamma.\mathsf{pk}$)), $\Gamma.\mathsf{encrypt}$ and $\Gamma.\mathsf{decrypt}$. Note that the encapsulation of the key used to encrypt a given string is always contained in the ciphertext.

We assume for simplicity that the space of signatures produced by $\Sigma$ is the same as the space of messages encrypted by $\Gamma$.

Let $m \in \{0,1\}^\star$ be a message, we propose the following scheme:

**Setup.** Invoke $\Gamma.\mathsf{setup}$ and $\Sigma.\mathsf{setup}$.

**Key Generation.** Invoke $\Sigma.\mathsf{keygen}$ and $\Gamma.\mathsf{keygen}$ to generate $\Sigma.\mathsf{sk}$, $\Sigma.\mathsf{pk}$ $\Gamma.\mathsf{sk}$ and $\Gamma.\mathsf{pk}$. Set the public key to $(\Sigma.\mathsf{pk}, \Gamma.\mathsf{pk})$ and the private key to ($\Sigma.\mathsf{sk}$, $\Gamma.\mathsf{sk}$).

**Signature.** First compute an encapsulation $c$ together with its decapsulation $k$ using $\Gamma.\mathsf{pk}$. Then compute a (digital) signature $\sigma = \Sigma.\mathsf{sign}_{\Sigma.\mathsf{sk}}(m\|c)$ on $m\|c$. Finally encrypt the resulting signature under $\Gamma.\mathsf{pk}$ (using $k$). Output $\mu = \Gamma.\mathsf{encrypt}_{\Gamma.\mathsf{pk}}(\sigma)$. Note that $c$ is part of $\mu$.

**Verification (By the Signer.)** To check the validity of an undeniable signature $\mu$ (that comprises the encapsulation $c$), issued on a certain message $m$, the signer first computes $\sigma = \Gamma.\mathsf{decrypt}_{\Gamma.\mathsf{sk}}(\mu)$, then calls $\Sigma.\mathsf{verify}$ on $\sigma$ and $m\|c$ using $\Sigma.\mathsf{pk}$. $\mu$ is valid if and only if the output of the latter item is 1.

**Confirmation/Denial Protocol.** To confirm (deny) a purported signature $\mu$ (containing the encapsulation $c$) on a certain message $m$, the signer first computes $\sigma = \Gamma.\mathsf{decrypt}_{\Gamma.\mathsf{sk}}(\mu)$, then invokes the algorithm $\Sigma.\mathsf{verify}$ on $\sigma$ and $m\|c$. According to the result, the signer issues a proof of knowledge of the decryption of $\mu$ that passes (does not pass) the verification algorithm $\Sigma.\mathsf{verify}$.

**Universal Conversion.** Release $\Gamma.\mathsf{sk}$.

### 4.3 Security Analysis and Efficiency Considerations

We first note that the properties of completeness, soundness and non-transferability of the confirmation/denial protocols are met by our construction as a direct consequence of the zero-knowledge proofs of knowledge. In the sequel, we prove that the construction resists existential forgeries and that signatures are invisible.

**Theorem 1.** *Our generic construction is $(t, \epsilon, q_s)$-EUF-CMA secure if the underlying digital signature scheme is $(t, \epsilon, q_s)$-EUF-CMA secure.*

*Proof.* Let $\mathcal{A}$ be an attacker that $(t, \epsilon, q_s)$-EUF-CMA breaks the existential unforgeability of our construction. We will construct an adversary $\mathcal{R}$ that $(t, \epsilon, q_s)$-EUF-CMA breaks the underlying digital signature scheme:

**Key generation.** $\mathcal{R}$ gets the parameters of the signature scheme in question from his challenger. Then he chooses an appropriate cryptosystem $\Gamma$ (obtained from the encryption of a signature paradigm) with parameters $\Gamma.\mathsf{pk}$, $\Gamma.\mathsf{sk}$, $\Gamma.\mathsf{encrypt}$ and $\Gamma.\mathsf{decrypt}$. $\mathcal{R}$ fixes the above parameters as a setting for the undeniable signatures $\mathcal{A}$ is trying to attack.

**Signature queries.** For a signature query on a message $m$, $\mathcal{R}$ will first compute an encapsulation $c$ together with its decapsulation $k$ (using $\Gamma.\mathsf{pk}$). Then he will request his challenger for a digital signature $\sigma$ on $m\|c$. Finally, he will encrypt $\sigma$ under $\Gamma.\mathsf{pk}$ (using $k$) and output the result to $\mathcal{A}$.

**Final Output.** Once $\mathcal{A}$ outputs his forgery $\mu^\star$ on $m^\star$. $\mathcal{R}$ will decrypt the signature to obtain $\sigma^\star$. If $\mu^\star$ is valid then by definition $\sigma^\star$ is valid too. $\mathcal{R}$ will output $\sigma^\star$ as a forgery on the message $(m^\star\|c^\star)$ where $c^\star$ is the encapsulation of the key that was used to encrypt $\sigma^\star$. In fact the probability that $m^\star\|c^\star$ has been queried by $\mathcal{R}$ on a query $m_i\|c_i$ ($m_i \neq m^\star$) is negligeable since $c_i$ is obtained by $\mathcal{R}$ from a random process (the encapsulation algorithm).

Note that there will be no need to simulate the confirmation/denial oracles since $\mathcal{A}$ has the universal receipt $\Gamma.\mathsf{sk}$ allowing the verification of the signatures. □

**Theorem 2.** *Our proposed construction is $(t, \epsilon, q_s, q_v)$-INV-CMA secure if it is $(t, \epsilon', q_s)$-EUF-CMA secure and the KEM used in the underlying cryptosystem is $(t + q_s q_v, \epsilon \cdot (1 - \epsilon')^{q_v})$-IND-CPA secure.*

*Proof.* Let $\mathcal{A}$ be an attacker that $(t, \epsilon, q_s, q_v)$-INV-CMA breaks our undeniable signatures, assumed to be $(t, \epsilon', q_s)$-EUF-CMA secure. We will construct an algorithm $\mathcal{R}$ that $(t + q_s q_v, \epsilon \cdot (1 - \epsilon')^{q_v})$-IND-CPA breaks the underlying KEM:

**Phase 1**

**Key Generation.** $\mathcal{R}$ gets the parameters of the KEM $\mathcal{K}$ from his challenger. Then he chooses an appropriate IND-OT secure DEM together with a signature scheme $\Sigma$.

**Signature Queries.** For a signature query on $m$. $\mathcal{R}$ first fixes a session key $k$ together with its decapsulation $c$ using $\mathcal{K}.pk$. Then he computes a (digital) signature $\sigma$ on $m\|c$ using $\Sigma.\mathsf{sk}$. Finally, he encrypts the produced signature (using $k$) and outputs the result to $\mathcal{A}$. $\mathcal{R}$ will maintain a list $\mathcal{L}$ of the queries he got (messages), the corresponding digital signatures and finally the signatures he issued.

**Verification (Confirmation/Denial) Queries.** For a signature $\mu$ on $m$, $\mathcal{R}$ will look up the list $\mathcal{L}$. If a record having as first component the message $m$ and third component $\mu$ appears in the list, then $\mathcal{R}$ will execute the confirmation protocol, otherwise, he will run the denial protocol. This simulation differs from the real one when the signature $\mu$ is valid and has not been obtained from a signature query. Thus, $\mu$ will correspond to a valid existential forgery of the undeniable signature scheme in question[4]. Hence, the probability that this scenario does not happen is at least $(1 - \epsilon')^{q_v}$ because the undeniable signature scheme is $(t, \epsilon', q_s)$-EUF-CMA secure by assumption. Finally, $\mathcal{R}$ can issue such proofs of knowledge, without knowing the private key of $\mathcal{K}$, using the rewinding technique because the protocols are zero knowledge, thus simulatable.

**Challenge.** Eventually, $\mathcal{A}$ outputs a challenging message $m^\star$. $\mathcal{R}$ will use his challenge $(c^\star, k^\star)$ to compute a digital signature using $\Sigma.\mathsf{sk}$ on $m^\star\|c^\star$. Then he encrypts the resulting signature using $k^\star$ and outputs the result $\mu^\star$ to $\mathcal{A}$. Therefore $\mu^\star$ is either a valid signature on $m^\star$ or a random element from the (undeniable) signatures space ($k^\star$ is random according to 2.4 and the DEM is IND-OT), which conforms to the game rules defined in 3.2.

**Phase 2** $\mathcal{A}$ will continue issuing queries to the signing, confirmation and denial oracles and $\mathcal{R}$ can answer as previously.

[4] This is the reason for generating a signature on the message in question concatenated with the encapsulation. In fact, valid signatures can only be obtained from the signing oracle (under the assumption that the scheme is EUF-CMA secure) even if the underlying cryptosystem offers the possibility of generating a different ciphertext for the same message (e.g., ElGamal [9]).

**Final Output**
When $\mathcal{A}$ outputs his answer $b \in \{0,1\}$, $\mathcal{R}$ will forward this answer to his own challenger. Therefore $\mathcal{R}$ will $(t + q_s q_v, \epsilon \cdot (1-\epsilon')^{q_v})$-IND-CPA break $\Gamma$. □

# 5 Construction of UCUS from Certain Pairing-Based Signatures Using KEMs

In the generic construction proposed in 4, the confirmation/denial protocols involve proofs of knowledge of the decryption of the undeniable signature and that this decryption is a digital signature on some known data. Therefore, one needs to consider a set of cryptosystems and signatures for which such proofs could be performed efficiently. One solution to achieve this is to consider the following class of signatures (KEMs).

## 5.1 Defining the Class $\mathbb{C}_1$ of Signatures and $\mathbb{K}$ of KEMs

**Definition 4.** $\mathbb{C}_1$ *is the set of pairing-based signatures such that:*

1. *The considered pairing $e$ is from $\mathbb{G} \times \mathbb{G}$ to $\mathbb{H}$.*
2. *The signature $\sigma$ on a message $m$ is written as $\sigma = (S, \bar{\sigma})$ such that*
   (a) *$\bar{\sigma} = \sigma \backslash S$ reveals no information about $m$ nor about $(\mathsf{sk}, \mathsf{pk})$ the key pair related to the given signature scheme.*
   (b) *$S \in \mathbb{G}$ and the verification equation of the signature is of the form: $e(S, P) = f(\bar{\sigma}, m, PP)$.*

   *where $P$ is a known generator of the group $\mathbb{G}$ (set as a public parameter of the scheme), $f$ is a public function, $m$ is the message in question and $PP$ are the known public parameters of the signature scheme*

The definition above may seem too restrictive but it already captures two very important pairing-based signatures, namely BLS [2] (where the message-key-independent part is the empty string) and Waters' [14] signatures.

**Definition 5.** $\mathbb{K}$ *is the set of KEMs such that:*

1. *The KEM is implemented in a bilinear group $\mathbb{G}$ where the considered pairing $e$ is from $\mathbb{G} \times \mathbb{G}$ to a group $\mathbb{H}$.*
2. *$P$ is a known generator of the group $\mathbb{G}$.*
3. *The session keys space $K$ is the same as the group $\mathbb{G}$.*
4. *Let $k \in \mathbb{G}$ be an element and $c$ a given encapsulation. On common input $e(k, P)$ and $c$:*
   - *If $k$ is the decapsulation of $c$, then there exists an efficient zero-knowledge proof $\mathcal{C}$ of this assertion, using the private key of the KEM,*
   - *otherwise, there exists an efficient zero-knowledge proof $\mathcal{D}$ of $k$ not being the decapsulation of $c$ (using also the private key of the KEM ).*

**A KEM in the Class $\mathbb{K}$:**

- $\mathsf{setup}$. Consider a bilinear group $\mathbb{G}$, with prime order $d$, generated by $P$.
- $\mathsf{keygen}$. Generate two values $x_1, x_2 \in \mathbb{Z}_d^\times$ and compute $X_1 = x_1 P$ and $X_2 = x_2 P$. Set the private key to $\mathsf{sk} = (x_1, x_2)$ and the public key to $\mathsf{pk} = (X_1, X_2)$.
- $\mathsf{encap}$. On input a nonce $(a, b) \in_R \mathbb{Z}_d^2$ and $\mathsf{pk}$, generate the *session key* $k = (a + b)P$ and its *encapsulation* $c = (aX_1, bX_2)$.
- $\mathsf{decap}$. Given $\mathsf{sk}$ and $c = (aX_1, bX_2)$, compute $k$ as $k = x_1^{-1} aX_1 + x_2^{-1} bX_2$.

This KEM is IND-CPA secure assuming the intractability of the *Decision Linear Problem.*

**Definition 6.** ***Decision Linear Problem (DLP).*** *Given $U$, $V$, $H$, $aU$, $bV$, $cH \in \mathbb{G}$, output 1 if $a + b = c \bmod (\#\mathbb{G})$ and 0 otherwise.*

The traditional DDH problem (corresponding to $b = 0$) can be reduced to DLP. In fact, DLP is believed to be hard even in bilinear groups where DDH is easy.

**Fact 1** *The KEM described above is in the class $\mathbb{K}$.*

*Proof.* – $X_1$ is a generator of $\mathbb{G}$.
– The proof $\mathcal{C}$ ($\mathcal{D}$) consists of the proof of equality (inequality) of the discrete logarithm of $X_2$ in base $P$ and of $e(bX_2, X_1)$ in base $e(k, X_1)e(aX_1, P)^{-1}$. We refer to [7] ([6]) for the proof of equality (inequality) of two discrete logarithms. □

### 5.2 Construction

Following the notations in 5.1 we consider an EUF-CMA digital signature scheme $\Sigma \in \mathbb{C}_1$ and an IND-CPA secure KEM $\mathcal{K} \in \mathbb{K}$, where the considered groups $\mathbb{G}$ and $H$, and the generator $P$ are the same for both $\Sigma$ and $\mathcal{K}$. We assume that the proofs $\mathcal{C}$ and $\mathcal{D}$ are known to the signer. A universally convertible undeniable signature, on a given message $m$, can be obtained by first invoking $\mathcal{K}$ to fix a key $k$ and its encapsulation $c$, then generating a digital signature $\sigma = (S, \bar{\sigma})$ on $m\|c$. The result is $\mu = (\mu_1, \mu_2, \mu_3) = (c, S + k, \bar{\sigma})$ [5]. Confirmation or denial of such a signature are achieved via the proofs $\mathcal{C}$ or $\mathcal{D}$ respectively, on the common input $m$, $\mu_1$ and $e(\mu_2, P)f(\mu_3, m\|c, PP)^{-1}$. In fact, if $k = \mathcal{K}.\mathsf{decap}(\mu_1)$ and $e(k, P) = e(\mu_2, P)f(\mu_3, m\|c, PP)^{-1}$, then the signer issues $\mathcal{C}$ (using the private key of the KEM). Otherwise, if $k = \mathcal{K}.\mathsf{decap}(\mu_1)$ and $e(k, P) \neq e(\mu_2, P)f(\mu_3, m\|c, PP)^{-1}$, he issues the proof $\mathcal{D}$. Finally, the universal conversion is done by releasing $\mathcal{K}.\mathsf{sk}$.

Unforgeability of such a construction is easily guaranteed by virtue of Theorem 1. As far as invisibility is concerned, we can base it directly on the underlying KEM. In fact, since $\bar{\sigma}$ does not reveal any information about the signing/verifying key (of the digital signature scheme) nor about the message in question, an attacker $\mathcal{A}$ capable of deciding on the validity of a given undeniable signature must definitely use information leaked by the encryption of the remaining part of the signature, that is $(c, k + S)$. Due to page limitation, the complete proofs will be given in the full version of the paper.

[5] The DEM encryption algorithm consists in adding the key to the message, whereas the decryption is the addition of the key inverse (in $\mathbb{G}$) to the ciphertext.

**Theorem 3.** *Let $\mathcal{A}$ be a $(t, \epsilon, q_s)$-EUF-CMA adversary against the above construction. Then, there exists a $(t, \epsilon, q_s)$-EUF-CMA adversary against the underlying digital signature scheme.* □

**Theorem 4.** *Our proposed construction is $(t, \epsilon, q_s, q_v)$-INV-CMA secure if it is $(t, \epsilon', q_s)$-EUF-CMA secure and the underlying KEM is $(t + q_s q_v, \epsilon.(1 - \epsilon')^{q_v})$-IND-CPA secure.* □

Instantiation of our framework with Waters' signatures [14] and the KEM described above results in a very efficient universally convertible undeniable signature scheme. In fact, the best scheme that was proposed so far [15] achieves the same security features (standard model and the same underlying standard assumptions), and thought it presents the additional quality of selective conversion, it has a longer signature and a higher signature generation and verification cost (approximately a multiplicative parameter $k$) and a higher key generation and universal conversion cost (a multiplicative parameter $2^{n/k}$), where $k$ is a public parameter to be optimized and $n$ is the length of the message to be signed.

## 6 Toward a Generic Construction of UCUS from Pairing-Based Signatures

In this section, we give the first generic construction of universally convertible undeniable signatures from a large class of pairing-based signatures, denoted $\mathbb{C}_2$, and from any IND-CPA cryptosystem whose decryption is efficiently verifiable.

### 6.1 Generic Construction

**Definition 7.** *$\mathbb{C}_2$ is the same set of signatures defined in Definition 4 with the exception of the verification equation being of the form $e(S, E) = f(\bar{\sigma}, m, PP)$, where $E \in \mathbb{G}$ is not necessarily a fixed generator of $\mathbb{G}$.*

It is clear that this class of signatures captures a large category of pairing-based signatures. In fact, almost all (pairing-based) signatures [2,1,16,14], that have been proposed so far, involve a pairing computation in the verification equation, between the key-message-dependent part of the signature and other entities. Note that the key-message-independent part in [2,16] is the empty string.

**Proposed Construction.** Let $\Sigma \in \mathbb{C}_2$ be an EUF-CMA signature from $\mathbb{C}_2$ and $\Gamma$ be an efficient decryption verifiable IND-CPA cryptosystem. Let further $d$ denote the group order of $\mathbb{G}$ and $p$ a suitable integer such that $\Gamma$ is IND-CPA secure in $\mathbb{Z}_p$ (the message space of $\Gamma$ is included in $\mathbb{Z}_p$). Note that $p > d$ due the contrast of key sizes between finite-field (or ring) and elliptic-curve cryptography.

We devise a universally convertible undeniable signature scheme as follows. First, we choose $r \in_R \mathbb{Z}_p$ then encrypt it under $\Gamma$ to result in $s = \Gamma.\mathsf{encrypt}_{\Gamma.\mathsf{pk}}(r)$. Then, generate a digital signature $(S, \bar{\sigma})$ on the message to be signed $m$ concatenated with $s$. The signature consists of the triple $\mu = (s, rS = (r \bmod d)S, \bar{\sigma})$.

To confirm (deny) a signature $\mu = (s, rS, \bar{\sigma})$, the signer decrypts $s$ then proves the equality (inequality) of the decryption of $s$ and the discrete logarithm of $e(rS, E)$ in base $f(\bar{\sigma}, m \| s, PP)$. Finally, the universal conversion is achieved by releasing $\Gamma.\mathsf{sk}$.

**Theorem 5.** *Let $\mathcal{A}$ be a $(t, \epsilon, q_s)$-EUF-CMA adversary against the above construction. Then, there exists a $(t, \epsilon, q_s)$-EUF-CMA adversary against the underlying digital signature scheme.* □

**Theorem 6.** *Our proposed construction is $(t, \epsilon, q_s, q_v)$-INV-CMA secure if it is $(t, \epsilon', q_s)$-EUF-CMA secure and the underlying cryptosystem is $(t + q_s q_v, \epsilon.(1 - \epsilon')^{q_v})$-IND-CPA secure.* □

Efficient realizations using this technique could be obtained by combining Waters' signatures [14] with an IND-CPA cryptosystem such as ElGamal [9] or Paillier [13].

## 7 Conclusion

In this paper, we proposed a construction for universally convertible undeniable signatures from secure digital signatures and some weakly secure cryptosystems. Next, we designed two efficient generic constructions for undeniable signatures from a large class of pairing-based signatures. These constructions found practical instantiations with some known signatures and cryptosystems. It might be good to analyze the security of the existing undeniable signature schemes or propose efficient ones using this technique. Finally, one is tempted to extend this approach to other "opaque" signatures such as directed signatures, or combine it with the techniques using commitment schemes in order to get better constructions.

## Acknowledgments

I would like to thank the anonymous reviewers for their helpful comments. Thanks go also to Joachim von zur Gathen for suggestions that improved the quality of the paper. This work was supported by the B-IT foundation.

## References

1. Boneh, D., Boyen, X.: Short Signatures Without Random Oracles. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 56–73. Springer, Heidelberg (2004)
2. Boneh, D., Lynn, B., Shacham, H.: Short Signatures from the Weil Pairing. J. Cryptology 17(4), 297–319 (2004)
3. Boyar, J., Chaum, D., Damgård, I.B., Pedersen, T.B.: Convertible undeniable signatures. In: Menezes, A.J., Vanstone, S.A. (eds.) CRYPTO 1990. LNCS, vol. 537, pp. 189–205. Springer, Heidelberg (1991)

4. Rackoff, C., Simon, D.R.: Non-Interactive Zero-Knowledge Proof of Knowledge and Chosen Ciphertext Attack. In: Feigenbaum, J. (ed.) CRYPTO 1991. LNCS, vol. 576, pp. 433–444. Springer, Heidelberg (1992)
5. Camenisch, J., Michels, M.: Confirmer Signature Schemes Secure against Adaptative Adversaries. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 243–258. Springer, Heidelberg (2000)
6. Camenisch, J., Shoup, V.: Practical Verifiable Encryption and Decryption of Discrete Logarithms. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 126–144. Springer, Heidelberg (2003)
7. Chaum, D., Pedersen, T.P.: Wallet Databases with Observers. In: Brickell, E.F. (ed.) CRYPTO 1992. LNCS, vol. 740, pp. 89–105. Springer, Heidelberg (1993)
8. Chaum, D., van Antwerpen, H.: Undeniable Signatures. In: Brassard, G. (ed.) CRYPTO 1989. LNCS, vol. 435, pp. 212–216. Springer, Heidelberg (1990)
9. El Gamal, T.: A Public Key Cryptosystem and a Signature Scheme based on Discrete Logarithms. IEEE Trans. Inf. Theory 31, 469–472 (1985)
10. Goldreich, O., Micali, S., Wigderson, A.: How to Prove all NP-Statements in Zero-Knowledge, and a Methodology of Cryptographic Protocol Design. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 171–185. Springer, Heidelberg (1987)
11. Herranz, J., Hofheinz, D., Kiltz, E.: KEM/DEM: Necessary and Sufficient Conditions for secure Hybrid Encryption (August 2006), `http://eprint.iacr.org/2006/265.pdf`
12. Laguillaumie, F., Vergnaud, D.: Short Undeniable Signatures Without Random Oracles: the Missing Link. In: Maitra, S., Veni Madhavan, C.E., Venkatesan, R. (eds.) INDOCRYPT 2005. LNCS, vol. 3797, pp. 283–296. Springer, Heidelberg (2005)
13. Paillier, P.: Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 223–238. Springer, Heidelberg (1999)
14. Waters, B.: Efficient Identity-Based Encryption Without Random Oracles. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 114–127. Springer, Heidelberg (2005)
15. Yuen, T., Au, M.H., Liu, J.K., Susilo, W. (Convertible) Undeniable Signatures Without Random Oracles. In: Qing, S., Imai, H., Wang, G. (eds.) ICICS 2007. LNCS, vol. 4861, pp. 83–97. Springer, Heidelberg (2007)
16. Zhang, F., Safavi-Naini, R., Susilo, W.: An Efficient Signature Scheme from Bilinear Pairings and Its Applications. In: Bao, F., Deng, R., Zhou, J. (eds.) PKC 2004. LNCS, vol. 2947, pp. 277–290. Springer, Heidelberg (2004)