Combined schemes for signature and encryption: the public-key and the identity-based setting

María Isabel González Vasco[*]
Florian Hess[†]
Rainer Steinwandt[‡]

Abstract

Consider a scenario in which parties use a public-key encryption scheme and a signature scheme with a single public key/private key pair—so the private key *sk* is used for both signing and decrypting. Such a simultaneous use of a key is in general considered poor cryptographic practice, but from an efficiency point of view looks attractive.

We offer security notions to analyze such violations of key separation. For both the identity- and the non-identity-based setting, we show that—although being insecure in general—for schemes of interest the resulting *combined scheme* can offer strong security guarantees.

Keywords: Combined scheme, identity-based cryptography, public-key cryptography, key separation.

1 Introduction

Using a single cryptographic key for different purposes is commonly considered poor cryptographic practice, as it violates the design principle of key separation. Notwithstanding this, already in the late late 90s Kelsey et al. [18] noted that there exist *forces pushing us toward a world in which different applications share common key material*: avoiding the cost for multiple certificates, (non-cryptographic) applications that simply default to a single user-specific key, and resource limitations on smart cards. For typical signature and public-key encryption schemes it may well happen that the secret-key-dependent operations are the very same—e. g., an exponentiation in a suitable group. If costly protection measures against side-channel or fault induction attacks need to be implemented, it is particularly tempting to work with a single key pair. Provided that there are no accidental interactions (in the sense of [18, Section 3]), one may hope for synergies in code size and implementation cost.

Haber and Pinkas [16] show that the simultaneous use of related keys in a signature scheme and a public-key encryption scheme is, for several examples, secure in a strong sense. They consider an adversary against a signature scheme which has unrestricted access to a decryption oracle of an encryption scheme using a related secret key, and prove that for several signature schemes such adversaries are not more

[*] Departamento de Matemática Aplicada, Universidad Rey Juan Carlos, c/ Tulipán, s/n, 28933 Madrid, Spain, mariaisabel.vasco@urjc.es

[†] Institut für Mathematik, Carl von Ossietzky Universität, 26111 Oldenburg, Germany, florian.hess@uni-oldenburg.de

[‡] Department of Mathematical Sciences, Florida Atlantic University, 777 Glades Road, Boca Raton, FL 33431, USA, rsteinwa@fau.edu

damaging than "standard" ones. Analogously, for some encryption schemes, they prove that an attacker who is granted unrestricted access to a signing oracle of a signature scheme using a related secret key will not endanger the security of the encryption scheme. Subsequent work focused on *universal padding schemes* that can be used for both signing and encryption without the need of separate keys. Coron et al. showed that PSS enables such a secure composition of a signature and encryption scheme with a single key pair [10]. More recently, Komano and Ohta propose combined constructions building on OAEP+ and REACT. Instead of the *partial-domain one-wayness* requirement of Coron et al., [19] imposes a one-wayness requirement only. Further refinements of universal paddings are explored by Chevallier-Mames et al. in [9].

Our contribution. Section 2 follows Haber and Pinkas [16] in the sense that we try to combine existing schemes that have not been designed for usage with a common private key. We analyze the security of such *combined schemes* using dedicated security notions building on the ones coined by Komano and Ohta in [19]. After showing how the simultaneous use of a private key can be fatal, we give a *combined scheme* with a security proof. This is constructed from the ElGamal signature scheme in the modification of Pointcheval and Stern [22] and an ElGamal encryption scheme under a Fujisaki-Okamoto conversion. We prove the resulting scheme to be secure in a strong sense: in the random oracle model, both existential unforgeability and indistinguishability of encryptions are achieved.

In the identity-based setting, working with a single user identity and one corresponding user key appears particularly natural, and Section 3 explores (for the first time in this context) the use of a unique private key in an identity-based setting: an identity-based encryption scheme and an identity-based signature scheme share a setup and key extraction algorithm and each user has one secret key only which is used for both signing and decrypting. We prove that such a simultaneous use can be possible without jeopardizing the security of the involved schemes. Namely, for an identity based signature scheme by Hess [17] and an identity based encryption scheme of Boneh and Franklin [8] we prove security in the sense of a natural generalization of standard security notions in identity-based cryptography.

Related (follow-up) work. In the years after making a preprint of our results available [23], some related work has appeared: the work of Degabriele et al. [11] on the EMV standards shows that EMV's RSA-based algorithms have security problems if a single key-pair is used for both signature and encryption; on the other hand, the elliptic curve algorithms that may end up as part of these standards are shown to be secure. Furthermore, in [20] Paterson et al. provide a way to construct a combined public-key scheme by means of an identity-based encryption scheme. They also offer a more efficient technique to obtain a combined public-key scheme, using the signature scheme of Boneh and Boyen [5] and an identity-based encryption scheme by the same authors [6].

If the essential application of an encryption and a signature scheme in a protocol consists of signing messages with a sender's private key followed by encrypting the signed messages under a recipient's public key, then signcryption [24] can be an alternative to separate encryption and signature mechanisms. As detailed in [1], a signcryption scheme induces a signature and an encryption scheme. With regard to key lengths, however, these induced schemes appear inferior to dedicated encryption or signature mechanisms, as essentially two signcryption keys are used to form one key for the induced signature or encryption scheme. For a scenario where we want the flexibility of separate encryption and signature mechanisms, the use of a signcryption scheme appears less attractive than a "secure key reuse" as described below. To find analogues to our security goals one would actually look at insider security against multi-user signcryption [1, 12] where both indistinguishability of ciphertexts and existential unforgeability must be achieved. More recently, Arriaga et al. [2] discussed *randomness reuse* when dealing with signcryption; their encrypt-then-sign and sign-then-encrypt constructions with randomness reuse are somewhat dual to the key reuse we consider. There, the key generations for the invoked encryption and signature schemes are independent, but the random coins used in the computation of a ciphertext and a signature coincide.

2 Combined public-key schemes

2.1 Preliminaries and definitions

Adapting the terminology from [16], we define a *combined public-key scheme* as a combination of a public-key encryption scheme and a signature scheme that have the key generation in common:

Definition 1 (combined public-key scheme)
A combined public-key scheme *is a tuple* $(\mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V})$ *of polynomial time algorithms:*

- $\mathcal{K}$ *is a probabilistic* key generation algorithm *that on input the security parameter* 1^k *outputs a public key/secret key pair* (pk, sk).
- $\mathcal{E}$ *is a probabilistic* encryption algorithm *that on input a message* m *and a public key* pk *computes a ciphertext* $c \leftarrow \mathcal{E}_{pk}(m)$.
- $\mathcal{D}$ *is a deterministic* decryption algorithm *that on input a candidate ciphertext* c *and a secret key* sk *outputs a plaintext* $m \leftarrow \mathcal{D}_{sk}(c)$ *or an error symbol* $\perp$.
- $\mathcal{S}$ *is a probabilistic* signing algorithm *that on input a message* m *and a secret key* sk *outputs a signature* $\sigma \leftarrow \mathcal{S}_{sk}(m)$.
- $\mathcal{V}$ *is a deterministic* verification algorithm *that on input a public key* pk*, a message* m *and a candidate signature* σ *outputs* true *or* false.

For a pair (pk, sk) *generated by* $\mathcal{K}$ *we require that with overwhelming probability the obvious correctness condition holds: For all messages* m *we have* $\mathcal{D}_{sk}(\mathcal{E}_{pk}(m)) = m$ *and* $\mathcal{V}_{pk}(m, \mathcal{S}_{sk}(m)) = \mathsf{true}$.

To model the security of a combined public-key scheme we adapt the notions of EUF-CCA2&ACMA and IND-CCA2&ACMA security introduced in [19]. For brevity, we deviate from [19] and simply write EUF-$\mathsf{CMA}^{\mathcal{D}}$ respectively $\mathsf{IND\text{-}CCA}^{\mathcal{S}}$ when considering "adversaries with an additional oracle." Essentially, the notion $\mathsf{IND\text{-}CCA}^{\mathcal{S}}$ formalizes the situation in which an IND-CCA adversary has, in addition to the usual tools, access to a signing oracle, and, analogously, an $\mathsf{EUF\text{-}CMA}^{\mathcal{D}}$ adversary is an EUF-CMA adversary having access to a decryption oracle, too. In particular, omitting $\mathcal{S}$, $\mathcal{V}$, and access to $\mathcal{O}_{\mathcal{V}}$ in Definition 2 yields the ordinary definition of a public-key encryption scheme offering ciphertext indistinguishability under chosen-ciphertext attacks (IND-CCA). Analogously, omitting $\mathcal{E}$, $\mathcal{D}$, and access to $\mathcal{O}_{\mathcal{D}}$ in Definition 3 yields the usual definition of a signature scheme that ensures existential unforgeability under chosen-message attacks (EUF-CMA).

Definition 2 ($\mathsf{IND\text{-}CCA}^{\mathcal{S}}$)
Let $(\mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V})$ *be a combined public-key scheme, and let* $\mathcal{A}$ *be a probabilistic polynomial time adversary. Consider the following attack scenario:*

1. *Compute a key pair* $(pk, sk) \leftarrow \mathcal{K}(1^k)$*, and hand* pk *as input to* $\mathcal{A}$.

2. *The adversary $\mathcal{A}$ is given unrestricted access to a signing oracle $\mathcal{O}_{\mathcal{S}}$ to run $\mathcal{S}_{sk}(\cdot)$ and unrestricted access to a decryption oracle $\mathcal{O}_{\mathcal{D}}$ to run $\mathcal{D}_{sk}(\cdot)$. At the end of this stage, $\mathcal{A}$ outputs two plaintexts $m_0 \neq m_1$ of equal length.*

3. *A value $b \in_R \{0,1\}$ is chosen uniformly at random, and $\mathcal{A}$ learns a target ciphertext $c \leftarrow \mathcal{E}_{pk}(m_b)$.*

4. *The algorithm $\mathcal{A}$ is again given unrestricted access to the signing oracle $\mathcal{O}_{\mathcal{S}}$, and the only restriction in querying $\mathcal{O}_{\mathcal{D}}$ is that target ciphertext c must not be queried. At the end of this stage $\mathcal{A}$ outputs a guess b' for b.*

The advantage $\mathrm{Adv}_{\mathcal{A}} = \mathrm{Adv}_{\mathcal{A}}(k)$ *of $\mathcal{A}$ is defined as $|2 \cdot P[b = b'] - 1|$, and we call* $(\mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V})$ secure in the sense of IND-CCA$^{\mathcal{S}}$ *if* $\mathrm{Adv}_{\mathcal{A}}$ *is negligible for all probabilistic polynomial time adversaries $\mathcal{A}$.*

Definition 3 (EUF-CMA$^{\mathcal{D}}$)
Let $(\mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V})$ be a combined public-key scheme, and let $\mathcal{A}$ be a probabilistic polynomial time adversary. Consider the following attack scenario:

1. *Compute a key pair $(pk, sk) \leftarrow \mathcal{K}(1^k)$, and hand pk as input to $\mathcal{A}$.*

2. *The adversary $\mathcal{A}$ is given unrestricted access to a signing oracle $\mathcal{O}_{\mathcal{S}}$ to run $\mathcal{S}_{sk}(\cdot)$ and unrestricted access to a decryption oracle $\mathcal{O}_{\mathcal{D}}$ to run $\mathcal{D}_{sk}(\cdot)$. At the end of this stage, $\mathcal{A}$ outputs a message m and a signature σ such that m has not been submitted to the signing oracle $\mathcal{O}_{\mathcal{S}}$.*

The success probability $\mathrm{Succ}_{\mathcal{A}} = \mathrm{Succ}_{\mathcal{A}}(k)$ *of $\mathcal{A}$ is defined as $P[\mathcal{V}_{pk}(m, \sigma) = \mathsf{true}]$, and we call* $(\mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V})$ secure in the sense of EUF-CMA$^{\mathcal{D}}$ *if* $\mathrm{Succ}_{\mathcal{A}}$ *is negligible for all probabilistic polynomial time adversaries $\mathcal{A}$.*

2.2 Combining secure schemes is not sufficient

As already indicated with the definitions in the previous section, our main focus is on the combination of IND-CCA-secure encryption and EUF-CMA-secure signature schemes. Before looking at this in more detail, we note that for passive adversaries the situation is somewhat trivial:

Remark 4 (IND-CPA+EUF-NMA) *Suppose we have a signature scheme that is secure against no message attacks (*EUF-NMA*), i. e., existentially unforgeable if the adversary has no access to a signing oracle, and a public-key encryption scheme that is secure against chosen plaintext attacks (*IND-CPA*), i. e., encryptions are indistinguishable in the absence of a decryption oracle. If these schemes have an identical key generation algorithm $\mathcal{K}$, then the resulting combined scheme certainly is secure against adversaries without access to a decryption or a signing oracle: the simultaneous use of the two schemes has no effect on the tools available to an adversary.*

For adversaries with access to stronger tools, different situations can arise: in the following example we see that combining an IND-CCA secure public-key encryption scheme and an EUF-CMA secure signature scheme is in general not sufficient to obtain a combined public-key scheme that is secure in the sense of IND-CCA$^{\mathcal{S}}$:

Example 5 (IND-CCA+EUF-CMA) *Given an* IND-CCA *secure public-key encryption scheme $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ we define a new scheme $(\mathcal{K}^*, \mathcal{E}^*, \mathcal{D}^*)$ where*

- *$\mathcal{K}^*$: outputs the same public key $pk^* := pk$ as $\mathcal{K}$, but the secret key $sk^* := (sk, r^*)$ contains, in addition to the secret key sk determined by $\mathcal{K}$, a random bitstring r^* of length linear in the security parameter and such that r^* cannot occur as encryption of any plaintext;*
- *$\mathcal{E}^*$: identical with $\mathcal{E}$;*
- *$\mathcal{D}^*$: checks if the ciphertext is equal to r^*. If yes, the secret value sk is returned, otherwise the algorithm $\mathcal{D}$ is applied.*

It is easy to see that $(\mathcal{K}^*, \mathcal{E}^*, \mathcal{D}^*)$ is secure in the sense of IND-CCA. *Now suppose we are also given an* EUF-CMA *secure signature scheme $(\mathcal{K}, \mathcal{S}, \mathcal{V})$. Using the algorithm $\mathcal{K}^*$ just defined we form a new signature scheme $(\mathcal{K}^*, \mathcal{S}^*, \mathcal{V}^*)$ where*

- *$\mathcal{S}^*$: runs $\mathcal{S}$ to obtain a signature σ, and outputs $\sigma^* = (\sigma, r^*)$, i. e., the secret bitstring r^* is appended to each signature.*
- *$\mathcal{V}^*$: on input $\sigma^* = (\sigma', r')$, outputs the same as $\mathcal{V}$ applied to σ'.*

Then $(\mathcal{K}^*, \mathcal{S}^*, \mathcal{V}^*)$ still offers security in the sense of EUF-CMA,[1] *but the combined public-key scheme $(\mathcal{K}^*, \mathcal{E}^*, \mathcal{D}^*, \mathcal{S}^*, \mathcal{V}^*)$ is clearly not secure in the sense of* IND-CCA$^{\mathcal{S}}$.

One would hope that problems as in the above (contrived) example do not occur in a "natural" setting; however, it is not always easy to argue formally whether given encryption/signature schemes can be combined securely. In the next section, we give an example of a combination of two established schemes resulting in EUF-CMA$^{\mathcal{D}}$ and IND-CCA$^{\mathcal{S}}$ security in the random oracle model. A central proof tool is the *Forking Lemma* as used by Pointcheval and Stern in [22].

2.3 A secure combined public-key scheme

We start by briefly describing the encryption scheme of Fujisaki and Okamoto based on ElGamal encryption and one-time padding [14], and the *Modified ElGamal (MEG)* signature scheme presented by Pointcheval and Stern in [22]. Moreover, we will summarize the essential results needed to prove the resulting combined public-key scheme secure in the sense of Definitions 2 and 3.

2.3.1 Fujisaki-Okamoto ElGamal based encryption

For any $k \in \mathbb{N}$ and for any $q \in \mathbb{N}$ of length k, let $H_1 : \{0, \dots, q-1\} \longrightarrow \{0,1\}^k$ and $H_2 : \{0, \dots, q-1\} \times \{0,1\}^k \longrightarrow \{0, \dots, q-1\}$ be random oracles. The encryption scheme described in Figure 1 was presented in [14] as an instantiation of the general Fujisaki-Okamoto conversion using ElGamal encryption as asymmetric component and a one-time pad as symmetric component.

It follows from the results of [14], that the above scheme is IND-CCA secure in the random oracle model, provided that the Decisional Diffie-Hellman assumption holds for G.

[1] Note, however, in case it was strongly unforgeable, it no longer is, as simply replacing r^* with a random string one can derive a new valid signature of the same message as in σ^*.

- $\mathcal{K}$: on input k it
  - randomly chooses a generator g of a cyclic group G of order q,
  - selects $x \in_R \{0, \ldots, q-1\}$ uniformly at random,
  - outputs (q, g, g^x) as public key and x as secret key;
- $\mathcal{E}$: on input $m \in \{0,1\}^k$, it
  - selects $r \in_R G$ uniformly at random,
  - computes $c_1 := r \cdot y^{H_2(r,m)}$, $c_2 := g^{H_2(r,m)}$ and $c_3 := H_1(r) \oplus m$,
  - outputs the ciphertext (c_1, c_2, c_3);
- $\mathcal{D}$: on input a ciphertext (c_1, c_2, c_3), it
  - computes $\hat{r} = c_1(c_2^x)^{-1}$,
  - retrieves $\hat{m} = H_1(\hat{r}) \oplus c_3$,
  - checks whether $c_1 = \hat{r}y^{H_2(\hat{r},\hat{m})}$, if this check fails, it outputs $\perp$, otherwise, it outputs $\hat{m}$.

Figure 1: ElGamal encryption with Fujisaki-Okamoto conversion

2.3.2 Modified ElGamal signature scheme

In [21] Pointcheval and Stern give a general strategy for providing security proofs for signature schemes in the random oracle model. One of the most prominent instantiations of this framework is a modification of the ElGamal Signature scheme (MEG), which we describe in Figure 2. At this, for any $k \in \mathbb{N}$ and any prime $q \in \mathbb{N}$ of length polynomial in k, let us assume a random oracle $F : \{0,1\}^* \times G \longrightarrow \{0, \ldots, q-1\}$ is publicly known. The existential unforgeability of the MEG scheme was proven in [21, 22] as an application of the so-called *Forking Lemma.* We review here the basic results and ideas behind this proof, which will be needed in the sequel. For more details, we refer to the original papers.

Remark 6 *In [21, 22], the case $q = p - 1$ for an α-hard prime*[2] *p is considered. For our purposes, the case of q being prime is sufficient, as for the combination with the encryption scheme discussed in Section 2.3.1, we want to build on the Decisional Diffie Hellman assumption in the group G.*

Pointcheval and Stern's results from [21, 22] apply to a large class of signature schemes, matching the following pattern: a signature of a message m consists of three components—a "commitment element" σ_1 sent by the signer, a random oracle image of this commitment together with the message, $h := F(m, \sigma_1)$, and a third component σ_2 which links σ_1 and h together and should be hard to compute without the secret

[2] An α-hard prime number p is such that $p - 1 = qR$ with q prime and $R \leq \|p\|^\alpha$, where $\|p\|$ denotes the length of p.

- $\mathcal{K}$: on input k it
  - randomly chooses a generator g of a subgroup G of $(\frac{\mathbb{Z}}{p\mathbb{Z}})^{\times}$ of order q. We thus represent the elements of G as integers mod p,
  - selects $x \in_R \{0, \ldots, q-1\}$ uniformly at random,
  - outputs (q, g, g^x) with $(g, g^x) \in G^2$ as public key and x as secret key;
- $\mathcal{S}$: on input a message $m \in \{0,1\}^*$, it
  - selects $K \in_R (\frac{\mathbb{Z}}{q\mathbb{Z}})^{\times}$ uniformly at random and defines $r := g^K \in G$,
  - sets $h := F(m, r)$ and selects $s \in \{0, \ldots, q-1\}$ so that $g^h = (g^x)^r r^s$, that is, s is the solution of the linear equation $F(m, r) = xr + Ks \mod q$,
  - outputs the triplet (r, h, s);
- $\mathcal{V}$: on input a triplet (r, h, s), it
  - outputs `true` if and only if $r \in G$ and $g^h = y^r r^s \mod p$.

Figure 2: Pointcheval-Stern Modified ElGamal Signature

signing key. These *generic signature schemes* can actually be obtained from any three-pass honest-verifier zero-knowledge identification protocol, as proven by Fiat and Shamir in [13].

For such generic schemes, it is proven [22, Theorem 1] that if a passive adversary is able to produce a valid forgery he will also be able to output two valid related signature tuples $(m, \sigma_1, h, \sigma_2)$, $(m, \sigma_1, h', \sigma_2')$, where h and h' are distinct and constructed using *different* random oracles F and F'. This result is commonly addressed as *Forking Lemma*. For the scenario considered here, this result has very strong implications, namely in [22, Theorem 6] the authors prove that for MEG such two tuples provide a solution for the discrete logarithm problem on input (g, g^x), provided that p is an α-hard prime. Moreover, [22, Lemma 6] states that actually, for an α-hard prime p, the same reasoning can be applied in the case of an active adversary, as the signing oracle can be simulated without the secret key with an indistinguishable distribution. As a result, the existential unforgeability of the MEG scheme can be proven, in the random oracle model, under the discrete logarithm assumption.

2.3.3 ElGamal based combined public-key scheme

Joining the above ElGamal based encryption and signature schemes in the natural way, we obtain a combined public-key scheme $(\mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V})$, where

- $\mathcal{K}$: on input k, the key generation algorithm
  - chooses a random large α-hard prime p, of length polynomial in k (and larger than k),

  - randomly chooses a generator g of a subgroup G of $(\frac{\mathbb{Z}}{p\mathbb{Z}})^\times$ of order q,
  - selects $x \in_R \{0, \ldots, q-1\}$ uniformly at random,
  - outputs (q, g, g^x) as public key and x as secret key;

- $\mathcal{E}$, $\mathcal{D}$: exactly as defined in Figure 1;
- $\mathcal{S}$, $\mathcal{V}$: exactly as defined in Figure 2.

The following proposition establishes rather strong security guarantees for this combined public-key scheme:

Proposition 7 *If the Decisional Diffie-Hellman assumption holds for G, the above ElGamal based combined public-key scheme is* $\mathsf{IND\text{-}CCA}^{\mathcal{S}}$ *and* $\mathsf{EUF\text{-}CMA}^{\mathcal{D}}$ *secure in the random oracle model.*

Proof: We consider the attack scenarios described in Definitions 2 and 3: Assume the key generation algorithm has been executed and the corresponding public key (q, g, g^x) has been forwarded to the adversary $\mathcal{A}$. Thus, $\mathcal{A}$ has access to all public information and can execute the encryption algorithm of the Fujisaki-Okamoto public-key encryption scheme and the verification algorithm of the MEG signature scheme. In addition, $\mathcal{A}$ has access to oracles $\mathcal{O}_{\mathcal{D}}$ and $\mathcal{O}_{\mathcal{S}}$ subject to the restrictions from Definition 2 respectively Definition 3.

$\mathsf{EUF\text{-}CMA}^{\mathcal{D}}$ *security.* We start by arguing that $\mathcal{O}_{\mathcal{S}}$ is not needed by $\mathcal{A}$, with a similar argument as in [22, Lemma 6]. Let `SSim` be a simulator that on input any message m outputs a valid signature triple (σ_1, h, σ_2), such that `SSim`'s output distribution is indistinguishable from the output distribution of the signature algorithm. Such a simulator exists, as proven in [22, Lemma 6] for α-hard prime numbers. At this, it is important to note that `SSim` does not hold the secret signing key.

Our adversary $\mathcal{A}$ will only notice that he is interacting with `SSim` instead of $\mathcal{O}_{\mathcal{S}}$ provided that one the following events occurs:

- E_1: there exists a triple (r, h, s) output by the simulator on input a message m, so that at some point $\mathcal{A}$ submitted the query (m, r) to the random oracle F; or
- E_2: there exist two triples (r, h, s), (r, h', s'), with $h \neq h'$, output by the simulator when queried (twice) with the same input m.

Note that $\mathcal{A}$ will only derive distinguishing information from the encryption algorithm or from interacting with $\mathcal{O}_{\mathcal{D}}$ if at some point he queries the random oracle F; thus, this event is captured by event E_1 above.

Now, as argued in the proof of [22, Lemma 4], the probability of events E_1 and E_2 together is, up to a constant factor, bounded by the probability of success of

$\mathcal{A}$ in making a forgery. As a result, we know that if $\mathcal{A}$ is able to create a forgery by querying $\mathcal{O}_{\mathcal{S}}$ with non-negligible probability, he will also be able to make a forgery interacting with `SSim` with non-negligible probability.

Thus, from now on we may assume the adversary to be equipped with `SSim` and $\mathcal{O}_{\mathcal{D}}$. It is easy to see that just as in the proof of Theorem 6.1 of [14] the decryption oracle can be simulated without the secret key. Let `DSim` be a simulator defined as in the aforementioned proof.

We can argue that $\mathcal{A}$ is not able to forge a signature with non-negligible probability interacting with `SSim` and `DSim` : indeed, the *Forking Lemma* [22, Theorem 1] guarantees that such an adversary that is able to forge a signature with non-negligible probability, can also solve the underlying discrete logarithm problem in polynomial time. In other words, our adversary $\mathcal{A}$, interacting with `SSim` and `DSim` (which hold no secret key) could decrypt arbitrary ciphertexts, which contradicts the $\mathsf{IND\text{-}CCA}$ security of the Fujisaki-Okamoto conversion. As a result, $\mathcal{A}$ cannot succeed when aiming at a forgery, and we have established $\mathsf{EUF\text{-}CMA}^{\mathcal{D}}$ security for the combined scheme.

$\mathsf{IND\text{-}CCA}^{\mathcal{S}}$ *security.* As we have argued that the signing oracle can be simulated without the secret key, our adversary is nothing more than a standard $\mathsf{IND\text{-}CCA}$ adversary against an $\mathsf{IND\text{-}CCA}$ secure scheme obtained with a Fujisaki-Okamoto conversion. Consequently, his advantage against this encryption scheme is negligible, and we see that the combined scheme is also $\mathsf{IND\text{-}CCA}^{\mathcal{S}}$ secure. qed.
□

3 Identity-based signature and encryption

In the context of identity-based cryptography, it appears natural to use the same identity for both encryption and signature purposes. One could explore the setting where a single key generation center is used for extracting signing and private decryption keys from user identities, and there is only a single set of public parameters; here we go one step further and consider, analogously as in the previous section, a situation in which each user has only one secret key to do both decrypting and signing. Clearly, this combination may actually yield a trivially insecure scheme (for instance, take an identity-based version of Example 5).

3.1 Preliminaries and definitions

To transfer the definition of a *combined public-key scheme* to the identity-based setting, we introduce the following definition:

Definition 8 (Combined identity-based scheme)
A combined identity-based scheme *is a tuple $(\mathcal{I}, \mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V})$ of polynomial time algorithms:*

- *$\mathcal{I}$ is a probabilistic* setup algorithm *that on input the security parameter 1^k returns public system parameters pk and a secret master key sk.*
- *$\mathcal{K}$ is a probabilistic* key extraction algorithm *which on input the public system parameters pk, the master key sk, and an identity $id \in \{0,1\}^*$ outputs a secret key sk_{id}, which can both be used for signature and decryption.*
- *$\mathcal{E}$ is a probabilistic* encryption algorithm *that on input the public parameters pk, an identity id and a plaintext m computes a ciphertext $c \leftarrow \mathcal{E}_{id,pk}(m)$.*
- *$\mathcal{D}$ is a deterministic* decryption algorithm *that on input a candidate ciphertext c, the public parameters pk and a secret key sk_{id} outputs a plaintext $m \leftarrow \mathcal{D}_{sk_{id},pk}(c)$ or an error symbol $\perp$.*
- *$\mathcal{S}$ is a probabilistic* signing algorithm *that on input a message m, public parameters pk and a secret key sk_{id} outputs a signature $\sigma \leftarrow \mathcal{S}_{sk_{id},pk}(m)$.*
- *$\mathcal{V}$ is a deterministic* verification algorithm *that on input the public parameters pk, an identity id, a message m and a candidate signature σ outputs* true *or* false*.*

For a pair (pk, sk) generated by $\mathcal{K}$ we require that with overwhelming probability the obvious correctness condition holds for all private keys sk_{id}: For all messages m we have $\mathcal{D}_{sk_{id},pk}(\mathcal{E}_{id,pk}(m)) = m$ and

$$\mathcal{V}_{id,pk}(m, \mathcal{S}_{sk_{id},pk}(m)) = \mathsf{true}.$$

Here the probability is taken over the random choices of $\mathcal{I}$, $\mathcal{K}$, $\mathcal{E}$, and $\mathcal{S}$.

To define the security of a combined identity-based scheme we adapt Definitions 2 and 3 accordingly, granting an adversary the (restricted) capability to obtain private keys, signatures and decryptions for identities of his choice:

Definition 9 ($\mathsf{IND\text{-}ID\text{-}CCA}^{\mathcal{S}}$)
Let $(\mathcal{I}, \mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V})$ be a combined identity-based scheme, and let $\mathcal{A}$ be a probabilistic polynomial time adversary. Consider the following attack scenario:

1. *Compute a key pair $(pk, sk) \leftarrow \mathcal{I}(1^k)$, and hand pk as input to $\mathcal{A}$.*
2. *The adversary $\mathcal{A}$ is given unrestricted access to a key extraction oracle $\mathcal{O}_{\mathcal{K}}$ to extract private keys, unrestricted access to a signing oracle $\mathcal{O}_{\mathcal{S}}$ and unrestricted access to a decryption oracle $\mathcal{O}_{\mathcal{D}}$.*[3] *At the end of this stage, $\mathcal{A}$ outputs two plaintexts $m_0 \neq m_1$ of equal length and an identity id_0 such that $\mathcal{O}_{\mathcal{K}}$ has not been queried for the corresponding secret key sk_{id_0}.*
3. *A value $b \in_R \{0, 1\}$ is chosen uniformly at random, and $\mathcal{A}$ learns a target ciphertext $c \leftarrow \mathcal{E}_{id_0,pk}(m_b)$.*
4. *The algorithm $\mathcal{A}$ is again given access to the key extraction oracle $\mathcal{O}_{\mathcal{K}}$, the signing oracle $\mathcal{O}_{\mathcal{S}}$, and the decryption oracle $\mathcal{O}_{\mathcal{D}}$, the only restrictions being that $\mathcal{O}_{\mathcal{D}}$ must not be queried to decrypt the target ciphertext c under sk_{id_0} and $\mathcal{O}_{\mathcal{K}}$ must not be queried for sk_{id_0}. At the end of this stage $\mathcal{A}$ outputs a guess b' for b.*

The advantage $\mathrm{Adv}_{\mathcal{A}} = \mathrm{Adv}_{\mathcal{A}}(k)$ *of $\mathcal{A}$ is defined as $|2 \cdot P[b = b'] - 1|$, and we call* $(\mathcal{I}, \mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V})$ secure in the sense of $\mathsf{IND\text{-}ID\text{-}CCA}^{\mathcal{S}}$ *if* $\mathrm{Adv}_{\mathcal{A}}$ *is negligible for all probabilistic polynomial time adversaries $\mathcal{A}$.*

Definition 10 ($\mathsf{EUF\text{-}ID\text{-}CMA}^{\mathcal{D}}$)
Let $(\mathcal{I}, \mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V})$ be a combined identity-based scheme, and let $\mathcal{A}$ be a probabilistic polynomial time adversary. Consider the following attack scenario:

1. *Compute a key pair $(pk, sk) \leftarrow \mathcal{I}(1^k)$, and hand pk as input to $\mathcal{A}$.*
2. *The algorithm $\mathcal{A}$ is given unrestricted access to a key extraction oracle $\mathcal{O}_{\mathcal{K}}$ to extract private keys, unrestricted access to a signing oracle $\mathcal{O}_{\mathcal{S}}$ and unrestricted access to a decryption oracle $\mathcal{O}_{\mathcal{D}}$. At the end of this stage, $\mathcal{A}$ outputs a message m, an identity id_0 and a signature σ such that $\mathcal{O}_{\mathcal{S}}$ has not been queried for a signature on m under sk_{id_0} and such that $\mathcal{O}_{\mathcal{K}}$ has not been queried for sk_{id_0}.*

The success probability $\mathrm{Succ}_{\mathcal{A}} = \mathrm{Succ}_{\mathcal{A}}(k)$ *of $\mathcal{A}$ is defined as $P[\mathcal{V}_{id_0,pk}(m, \sigma) = \mathsf{true}]$, and we call* $(\mathcal{I}, \mathcal{K}, \mathcal{E}, \mathcal{D}, \mathcal{S}, \mathcal{V})$ secure in the sense of $\mathsf{EUF\text{-}ID\text{-}CMA}^{\mathcal{D}}$ *if* $\mathrm{Succ}_{\mathcal{A}}$ *is negligible for all probabilistic polynomial time adversaries $\mathcal{A}$.*

[3] Signing and decryption oracles can access the master key sk and the public parameters pk, and hence can answer signature and decryption queries for arbitrary identities; that is, queries for these oracles are of the form $(id, message)$. For a fixed identity id queries are replied using always the same secret key sk_{id} (cf. the Init oracle in [3]).

Setup algorithm $\mathcal{I}$: chooses a random generator $P \in_R G$. Moreover, $Y_{\text{master}} := sk \cdot P$ is published, where $sk \in_R \left(\frac{\mathbb{Z}}{l\mathbb{Z}}\right)^{\times}$ is the uniformly at random chosen master key.

Key extraction $\mathcal{K}_{\mathbf{dec}}$: for an identity $id \in \{0,1\}^*$, the secret key is $sk_{id} := sk \cdot Q_{id}$, where $Q_{id} := H_1(id) \in G^*$.

Encryption algorithm $\mathcal{E}$: to encrypt a message $m \in \{0,1\}^n$ under the identity id, the following steps are performed:

- compute $Q_{id} := H_1(id) \in G^*$
- choose a random $\sigma \in_R \{0,1\}^n$ and set $r := H_3(\sigma, m) \in \left(\frac{\mathbb{Z}}{l\mathbb{Z}}\right)^{\times}$
- compute $g_{id} := e(Q_{id}, Y_{\text{master}}) \in V$

The ciphertext is $c := (r \cdot P, \sigma \oplus H_2(g_{id}^r), m \oplus H_4(\sigma))$.

Decryption algorithm $\mathcal{D}$: To decrypt a candidate ciphertext $c = (U, v, w)$ with secret key sk_{id}, the subsequent steps are performed:

- if $U \notin G^*$, the error symbol $\perp$ is returned
- compute $\sigma := v \oplus H_2(e(sk_{id}, U))$
- let $m := w \oplus H_4(\sigma)$ and $r := H_3(\sigma, m)$
- if $U \neq r \cdot P$ return the error symbol $\perp$, otherwise output m as decryption of c

Figure 3: Identity-based encryption from [8]

3.2 A secure combined identity-based scheme

To obtain an example of a secure combined identity-based scheme in the above setting, we use an identity-based encryption scheme of Boneh and Franklin [8] and combine it with an identity-based signature scheme proposed by Hess [17].

3.2.1 The FullIdent scheme of Boneh and Franklin

In [8] an identity-based encryption scheme is proposed that can be considered as the first practical proposal in this line of research and is referred to as FullIdent. It is similar to the IND-CCA secure ElGamal variation discussed in the previous section, in the sense that chosen ciphertext security is again obtained by means of the conversion technique of Fujisaki and Okamoto from [14]. For a detailed discussion of FullIdent and a proof of its IND-ID-CCA security we refer to [8, 15].

Consider $(G, +)$ and $(V, \cdot)$ two cyclic groups of prime order l, where $l = \Theta(2^k)$. Let $e : G \times G \to V$ be a suitable bilinear pairing. We write $G^* := G \setminus \{0\}$ and $V^* := V \setminus \{1\}$. Furthermore, consider four hash functions H_1, H_2, H_3, and H_4 of appropriate domain and range. With these ingredients, the scheme FullIdent is described in Figure 3.

Setup algorithm $\mathcal{I}$: chooses a random generator $P \in_R G$. Moreover, $Y_{\text{master}} := sk \cdot P$ is published, where $sk \in_R \left(\frac{\mathbb{Z}}{l\mathbb{Z}}\right)^{\times}$ is the uniformly at random chosen master secret key.

Key extraction $\mathcal{K}_{\mathbf{sig}}$: for an identity $id \in \{0,1\}^*$, the secret key corresponding to this identity is computed from the secret master key by the issuing authority as $sk_{id} := sk \cdot H(id)$ and forwarded to the signer.

Signing algorithm $\mathcal{S}$: to sign a message m with sk_{id}, the signer chooses arbitrary $P_1 \in G^*$, picks a random integer $k \in \left(\frac{\mathbb{Z}}{l\mathbb{Z}}\right)^{\times}$ and computes:

- $r := e(P_1, P)^k$
- $v := h(m, r)$
- $u := v \cdot sk_{id} + k \cdot P_1$

The signature on m under sk_{id} then is $(u, v) \in G \times \left(\frac{\mathbb{Z}}{l\mathbb{Z}}\right)^{\times}$.

Verification algorithm $\mathcal{V}$: returns true if and only if a candidate signature (u, v) for a message m satisfies the equation $v = h(m, r)$, where r is computed as $r = e(u, P) \cdot e(H(id), -Y_{\text{master}})^v$.

Figure 4: Identity-based signature scheme from [17]

3.2.2 Hess' identity-based signature scheme

Consider, as above, $(G, +)$ and $(V, \cdot)$ two cyclic groups of prime order l, where $l = \Theta(2^k)$. Let $e : G \times G \to V$ be a suitable bilinear pairing. Furthermore, consider two hash functions

$$\begin{array}{lcl} h : \{0,1\}^* \times V & \longrightarrow & (\mathbb{Z}/l\mathbb{Z})^{\times} \quad \text{and} \\ H : \{0,1\}^* & \longrightarrow & G^*. \end{array}$$

The EUF-ID-CMA security of the scheme in Figure 4, in the random oracle model, relies on the hardness of the Computational Diffie Hellman Problem in G, as proven in [17, Theorem 1].

3.2.3 Combining FullIdent with Hess' signature scheme

The identity-based schemes just described both use a discrete logarithm sk of a public

$$Y_{\text{master}} = sk \cdot P$$

as secret master key. Moreover, also the key extraction algorithm is identical. Thus, it seems natural to form a combined identity-based scheme with the same setup and key extraction algorithm.

As public parameters pk we use a single value $Y_{\text{master}} = sk \cdot P \in G$ along with all the remaining (sk-independent) needed public parameters—like P and the specification of the random oracles $H = H_1, H_2, H_3, H_4, h$—and of, only one, admissible bilinear map e. Note that both schemes will make use of the hash function H and of the bilinear map e; moreover, we shall assume that all these involved hash functions behave like independent random oracles.

Proposition 11 *In the random oracle model, if the Bilinear Diffie Hellman assumption for (G, e) holds, then the above combined identity-based scheme is secure in the sense of* IND-ID-CCA$^{\mathcal{S}}$ *and in the sense of* EUF-ID-CMA$^{\mathcal{D}}$*.*

Proof: We show that a successful adversary in the sense of $\mathsf{IND\text{-}ID\text{-}CCA}^{\mathcal{S}}$ yields an $\mathsf{IND\text{-}ID\text{-}CCA}$ adversary against Boneh and Franklin's $\mathsf{FullIdent}$ scheme. Similarly, we argue that a successful $\mathsf{EUF\text{-}ID\text{-}CMA}^{\mathcal{D}}$ adversary would break the $\mathsf{EUF\text{-}ID\text{-}CMA}$ security of Hess' signature scheme.

$\mathsf{IND\text{-}ID\text{-}CCA}^{\mathcal{S}}$ *security.* Suppose we have an adversary $\mathcal{A}$ in the sense of $\mathsf{IND\text{-}ID\text{-}CCA}^{\mathcal{S}}$ violating ciphertext indistinguishability. We start by showing that the signing oracle $\mathcal{O}_S$ can be simulated for each identity without the corresponding secret key; thus, it is of no help to $\mathcal{A}$.

Consider a simulator `SSim` which on input of an identity id and a message m chooses $(u, v) \in_R G \times \left(\frac{\mathbb{Z}}{l\mathbb{Z}}\right)^{\times}$ and defines the hash value $h(m, r) := v$ where $r = e(u, P) \cdot e(H(id), -Y_{\text{master}})^v$. The output of `SSim` is (u, v) which is a valid signature of m under id. The running time of `SSim` is comprised of the running time of the verification step (and of the book keeping for H and h) and is thus polynomial in the running time of $\mathcal{A}$.

This simulator `SSim` can only be distinguished from a true signing oracle if at some point a queried hash value $h(m, r)$ is already defined. The probability for this is $O(q_S^2/2^k)$ if at most most q_S signing queries are issued to `SSim`. Since $\mathcal{A}$ and thus q_S is polynomial in k, this probability is negligible.

From the above observations we see that an $\mathsf{IND\text{-}ID\text{-}CCA}^{\mathcal{S}}$ adversary $\mathcal{A}$ violating ciphertext indistinguishability can be transformed into an ordinary $\mathsf{IND\text{-}ID\text{-}CCA}$ attacker against the Boneh-Franklin scheme.

$\mathsf{EUF\text{-}ID\text{-}CMA}^{\mathcal{D}}$ *security.* Now suppose that an $\mathsf{EUF\text{-}ID\text{-}CMA}^{\mathcal{D}}$ adversary $\mathcal{A}$ successfully violates the existential unforgeability of the combined identity-based scheme in question.

Again, we argue that a decryption oracle can be simulated without the corresponding identities' private keys: such a simulator `DSim` exists for the full encryption scheme of [7]. A general argument for this is that the scheme arises as the Fujisaki-Okamoto transform of a γ-uniform and $\mathsf{ID\text{-}OWE}$ secure encryption scheme, see [7] and [14, Corollary 13]. In the following we give a more specific argument.

Valid ciphertexts are of the form

$$(rP, \sigma \oplus H_2(e(H(id), Y_{\text{master}})^r), m \oplus H_4(\sigma))$$

where H_2, H_3, H_4 are suitable fixed hash functions, σ, m are bitstrings and the equation $r = H_3(\sigma, m)$ holds true. Note that the map

$$f : (r, \sigma, m) \mapsto (rP, \sigma \oplus H_2(e(H(id), Y_{\text{master}})^r), m \oplus H_4(\sigma))$$

is bijective. The decryption function $\mathcal{D}$ inverts it internally to (σ, m) using the secret key, and returns m as the message if $r = H_3(\sigma, m)$ holds true, and the error symbol $\perp$ otherwise.

The simulator `DSim` is defined as follows. First, the simulator `DSim` is arranged to know the queries to the oracle of the hash function H_3. Second, for a decryption query on a given ciphertext c it checks whether one of the queries (σ, m) passed to the oracle of H_3 so far satisfies $f(r, \sigma, m) = c$ with $r = H_3(\sigma, m)$. If yes, the answer to the decryption query is m, otherwise the error symbol $\perp$.

We now compare two runs of $\mathcal{A}$ with identical inputs, in particular with identical random tapes and hence identical hash functions H, H_2, H_3, H_4, but in one case with access to the decryption function $\mathcal{D}$ and in the other case with access to the simulator `DSim`. Obviously $\mathcal{A}$ will execute identically until the, say, i-th decryption query will be answered differently by $\mathcal{D}$ and `DSim`. If `DSim` outputs the message m on input of c, then $r = H_3(\sigma, m)$ for r, σ, m with $f(r, \sigma, m) = c$, and $\mathcal{D}$ also outputs the message m. A difference can hence only occur if `DSim` outputs $\perp$ on input of c, and $\mathcal{D}$ outputs a message on input of the same c. This means that, while again $r = H_3(\sigma, m)$ for r, σ, m with $f(r, \sigma, m) = c$ must hold, the oracle for H_3 has not been queried for σ, m in the run of $\mathcal{A}$ with `DSim`. Then the oracle for H_3 has also not been queried for σ, m in the run of $\mathcal{A}$ with $\mathcal{D}$, since the runs are identical up to the i-th decryption query (note that only queries to the oracle of H_3 outside DSim and $\mathcal{D}$ count). Randomizing over the input of $\mathcal{A}$ and the hash functions we see that the probability of different outputs of DSim and $\mathcal{D}$ in the i-th decryption query is equal to the probability that $\mathcal{A}$ computes c such that $r = H_3(\sigma, m)$ for r, σ, m with $f(r, \sigma, m) = c$ without querying the oracle of H_3 for σ, m. Since H_3 assumes random values in $(\mathbb{Z}/l\mathbb{Z})^{\times}$, this probability is $1/(l-1)$.

Now, if $\mathcal{A}$ makes at most q_D decryption queries, then the probability that DSim answers all these decryption queries like $\mathcal{D}$ is $(1 - 1/(l-1))^{q_D}$. Since $l = \Theta(2^k)$ and q_D is polynomial in k, this probability differs only negligibly from 1.

The running time of DSim is essentially that of encryption times the number of queries to the oracle of H_3 and is thus polynomial in k.

From the above observations we see that an EUF-ID-CMA$^{\mathcal{D}}$ adversary $\mathcal{A}$ producing a forged signature can be transformed into an ordinary EUF-ID-CMA attacker against the scheme of Hess, which finishes the proof. qed.

□

4 Conclusions

Building on earlier work in [16, 19, 10], our discussion offers formal security notions to analyze combined schemes both in an identity-based and in a non-identity-based setting. We give two concrete constructions using established schemes and prove them secure.

For the non identity-based case, the *Forking Lemma* turned out to be a powerful tool: as in our example the signing oracle can be simulated without knowledge of the secret key, a successful adversary constructing an existential forgery would be able to efficiently solve the mathematical problem underlying both the signature and the encryption scheme. Similarly, for the identity-based setting, the strategy is to argue that both the signing and decryption oracle can be simulated without the corresponding secret key. As a result, the “combined” adversary reduces to a standard one and the security level is thus inherited from the constituent encryption and signature schemes. Aiming at the identification of further secure combined schemes, the above proof strategies appear to be quite promising.

Acknowledgments. The authors thank Nigel Smart for noticing the connection between independent prior work by the authors which led to this joint paper. Moreover, the authors are grateful for anonymous reviewer feedback that helped to improve the original manuscript. First and last author were supported by the Spanish *Ministerio de Economía y Competitividad* through the project grant MTM-2012-15167.

References

[1] J. H. An, Y. Dodis, and T. Rabin. On the Security of Joint Signature and Encryption. In Lars R. Knudsen, editor, *Advances in Cryptology – EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 83–107. Springer-Verlag, 2002.

[2] A. Arriaga, M. Barbosa, and P. Farshim. On the Joint Security of Signature and Encryption Schemes under Randomness Reuse: Efficiency and Security Amplification. In F. Bao, P. Samarati, and

J. Zhou, editors, *Applied Cryptography and Network Security – ACNS 2012*, volume 7341 of *Lecture Notes in Computer Science*, pages 206–223. Springer-Verlag, 2012.

[3] M. Bellare, C. Namprempere, and G. Neven. Security Proofs for Identity-Based Identification and Signature Schemes. Available at `http://www-cse.ucsd.edu/users/mihir/papers/ibi.html`, May 2004. Full version of [4].

[4] M. Bellare, C. Namprempere, and G. Neven. Security Proofs for Identity-Based Identification and Signature Schemes. In C. Cachin and J. Camenisch, editors, *Advances in Cryptology – EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 268–286. Springer-Verlag, 2004.

[5] D. Boneh and X. Boyen. Short Signatures Without Random Oracles and the SDH Assumption in Bilinear Groups. *Journal of Cryptology*, 21(2):149–177, 2008.

[6] D. Boneh and X. Boyen. Efficient Selective Identity-Based Encryption Without Random Oracles. *Journal of Cryptology*, 24(4):659–693, 2011.

[7] D. Boneh and M. Franklin. Identity-Based Encryption from the Weil Pairing. In J. Kilian, editor, *Advances in Cryptology – CRYPTO 2001*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer-Verlag, 2001.

[8] D. Boneh and M. Franklin. Identity-Based Encryption from the Weil Pairing. *SIAM J. of Computing*, 32(3):586–615, 2003. Extended abstract appeared in [7].

[9] B. Chevallier-Mames, D. Hieu-Phan, and D. Pointcheval. Optimal Asymmetric Encryption and Signature Paddings. In J. Ioannidis, A. Keromytis, and M.Yung, editors, *Applied Cryptography and Network Security – Proceedings of ACNS 2005*, volume 3531 of *Lecture Notes in Computer Science*, pages 254–268. Springer-Verlag, 2005.

[10] J.-S. Coron, M. Joye, D. Naccache, and P. Paillier. Universal Padding Schemes for RSA. In M. Yung, editor, *Advances in Cryptology – CRYPTO 2002*, volume 2442 of *Lecture Notes in Computer Science*, pages 226–241. Springer-Verlag, 2002.

[11] J.P. Degabriele, A. Lehmann, K.G. Paterson, N.P. Smart, and M. Strefler. On the Joint Security of Encryption and Signature in EMV. In O. Dunkelmann, editor, *Topics in Cryptology – CT-RSA 2012*, volume 7178 of *Lectures Notes in Computer Science*, pages 116–135. Springer-Verlag, 2012.

[12] Y. Dodis, M. J. Freedman, S. Jarecki, and S. Walfish. Versatile Padding Schemes for Joint Signature and Encryption. In *Proceedings of the 11th ACM Conference on Computer and Communications Security — CCS '04*, pages 344–353. ACM, 2004.

[13] A. Fiat and A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. In A. M. Odlyzko, editor, *Advances in Cryptology – Crypto '86*, volume 263 of *Lecture Notes in Computer Science*, pages 186–194. Springer-Verlag, 1987.

[14] E. Fujisaki and T. Okamoto. Secure Integration of Asymmetric and Symmetric Encryption Schemes. In M. Wiener, editor, *Advances in Cryptology – CRYPTO 1999*, volume 1666 of *Lecture Notes in Computer Science*, pages 537–554. Springer-Verlag, 1999.

[15] D. Galindo. Boneh-Franklin Identity Based Encryption Revisited. In L. Caires, G. F. Italiano, L. Monteiro, C. Palamidessi, and M. Yung, editors, *Automata, Languages and Programming, 32nd International Colloquium, ICALP 2005*, volume 3580 of *Lecture Notes in Computer Science*, pages 791–802. Springer-Verlag, 2005.

[16] S. Haber and B. Pinkas. Securely Combining Public-Key Cryptosystems. In *Proceedings of the 8th ACM conference on Computer and Communications Security*, pages 215–224. ACM, 2001.

[17] F. Hess. Efficient Identity based Signature Schemes based on Pairings. In K. Nyberg and H. Heys, editors, *Proceedings of SAC 2002*, volume 2595 of *Lecture Notes in Computer Science*, pages 310–324. Springer-Verlag, 2003.

[18] J. Kelsey, B. Schneier, and D. Wagner. Protocol Interactions and the Chosen Protocol Attack. In B. Christianson, B. Crispo, M. Lomas, and M. Roe, editors, *Proceedings of the 5th International Workshop on Security Protocols*, volume 1361 of *Lecture Notes in Computer Science*, pages 91–104. Springer-Verlag, 1998.

[19] Y. Komano and K. Ohta. Efficient Universal Padding Techniques for Multiplicative Trapdoor One-Way Permutation. In D. Boneh, editor, *Advances in Cryptology – CRYPTO 2003*, volume 2729 of *Lecture Notes in Computer Science*, pages 366–382. Springer-Verlag, 2003.

[20] K. G. Paterson, J. C.N. Schuldt, M. Stam, and S. Thomson. On the Joint Security of Encryption and Signature, Revisited. In D.H. Lee and X. Wang, editors, *Advances in Cryptology – ASIACRYPT 2011*, volume 7073 of *Lecture Notes in Computer Science*, pages 161–178. Springer-Verlag, 2011.

[21] D. Pointcheval and J. Stern. Security Proofs for Signature Schemes. In U. Maurer, editor, *Advances in Cryptology – EUROCRYPT '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 387–398. Springer-Verlag, 1996.

[22] D. Pointcheval and J. Stern. Security Arguments for Digital Signatures and Blind Signatures. *Journal of Cryptology*, 13(3):361–396, 2000.

[23] M.I. González Vasco, F. Hess, and R. Steinwandt. Combined (identity-based) public key schemes. Cryptology ePrint Archive, Report 2008/466, 2008. `http://eprint.iacr.org/2008/466`.

[24] Y. Zheng. Digital Signcryption or How to Achieve Cost(Signature & Encryption) << Cost(Signature) + Cost(Encryption). In B. S. Kaliski Jr., editor, *Advances in Cryptology – CRYPTO '97*, volume 1294 of *Lecture Notes in Computer Science*, pages 165–179. Springer-Verlag, 1997.