Efficient Constructions of Signcryption Schemes and Signcryption Composability

Takahiro Matsuda*, Kanta Matsuura, and Jacob C.N. Schuldt

The University of Tokyo, Japan
{tmatsuda,kanta,schuldt}@iis.u-tokyo.ac.jp

Abstract. In this paper, we investigate simple but efficient constructions of signcryption schemes. Firstly, we show how symmetric primitives can be used to efficiently achieve outsider multi-user security, leading to a signcryption scheme with the currently lowest ciphertext and computational overhead. For the mixed security notions outsider confidentiality/insider unforgeability and insider confidentiality/outsider unforgeability, this approach yields lower ciphertext overhead and a higher level of security, respectively, compared to the current schemes. Secondly, we show a simple optimization to the well known "sign-then-encrypt" and "encrypt-then-sign" approaches to the construction of signcryption schemes by using tag-based encryption. Instantiations with our proposed tag-based schemes yield multi-user insider secure signcryption schemes in the random oracle model which is at least as efficient as any other existing scheme both in terms of ciphertext overhead and computational cost. Furthermore, we show that very efficient standard model signcryption schemes can be constructed using this technique as well. Lastly, we show how signatures and encryption can be combined in a non-black-box manner to achieve higher efficiency than schemes based on the above approach. We refer to signature and encryption schemes which can be combined in this way as *signcryption composable*, and we show that a number of the most efficient standard model encryption and signature schemes satisfy this, leading to the most efficient standard model signcryption schemes. Since all of our constructions are fairly simple and efficient, they provide a benchmark which can be used to evaluate future signcryption schemes.

Keywords: signcryption, multi-user security, generic construction.

1 Introduction

The notion signcryption was introduced by Zheng [47] as a primitive providing the combined functionality of signatures and encryption i.e. unforgeability, message confidentiality, and possibly non-repudiation. The main motivation given in [47] for introducing signcryption as a new primitive was to achieve higher efficiency than simply combining signature and encryption. While the scheme

* Takahiro Matsuda is supported by a JSPS fellowship.

B. Roy and N. Sendrier (Eds.): INDOCRYPT 2009, LNCS 5922, pp. 321–342, 2009.

proposed in [47] was not formally proved secure, this was done in subsequent works [5,6]. Furthermore, An *et al.* [3] formally analyzed the security of the simple composition of signature and public key encryption (PKE).

Since the introduction of the primitive, many signcryption schemes have been proposed, e.g. [47,3,24,29,30,18,8,28,20,42,43,44]. However, these schemes provide different security levels depending on the used security model. The simplest security model for a signcryption scheme considers a two-user system consisting only of a single sender and a single receiver. While two-user security models have been considered in some of the earlier papers (e.g. [3,18]), they are of limited interest since most practical systems will include many users, and for signcryption schemes, two-user security does not imply multi-user security[1]. Another aspect of the security model is whether the adversary is considered to be an *insider*, possibly playing the part of either the sender or receiver, or an *outsider* trying to attack an uncompromised sender and receiver pair. Note that many schemes are proved secure using a "mix" of these security notions. e.g. insider confidentiality and outsider unforgeability [5,6], or outsider confidentiality and insider unforgeability [24,20]. The efforts to construct schemes providing security in the strongest sense, i.e. insider security for both confidentiality and unforgeability, have met some challenges. For example, the scheme proposed in [31] was shown to be insecure in [38,46], "fixed" in [46], only to be broken again in [39]. Finally, Libert *et al.* [29] updated the original scheme [31] while Li *et al.* [28] independently proposed a scheme based on [46], which both seem to be resistant to the attacks in [38,46,39]. In a similar way, the scheme proposed in [32] was shown to be insecure in [40], updated in [33], only to be shown insecure in [41]. Lastly, Libert *et al.* [30] updated the original scheme to be resistant to the attack in [40]. This illustrates that care must be taken when designing fully insider secure signcryption schemes.

Except the composition results by An *et al.* [3] and the relation between key agreement and signcryption key encapsulation mechanisms (signcryption KEMs) studied by Gorantla *et al.* [20], most constructions of signcryption schemes make very little use of existing primitives and the established security properties of these. Furthermore, the proposed signcryption schemes are rarely compared to the often simpler constructions of signcryption using existing primitives and the efficiency achieved by these. As the proposed constructions get increasingly complex, as in the case of the recently proposed standard model schemes [42,43,44], this leaves open the question whether the direct constructions provide any advantages compared to the signcryption schemes relying on other primitives, when these are instantiated properly.

Our Contribution. We focus on simple but efficient constructions of signcryption using existing primitives or simple extension of these. Firstly, we show how the

[1] E.g. see [6] for a discussion of this. Furthermore, note that An *et al.* [3] showed how a simple composition of signatures and encryption achieving two-user security can be transformed into a scheme achieving multi-user security, but this transformation is not applicable in general.

properties of symmetric key encryption (SKE) and message authentication codes (MAC) can be used to provide outsider security. As a tool, we use a tag-based non-interactive key exchange (TNIKE) scheme, which is a simple extension of an ordinary non-interactive key exchange (NIKE) scheme [19,15] and is easy to instantiate in the random oracle model. The resulting scheme has a lower computational cost and ciphertext overhead than any of the existing signcryption schemes. If insider unforgeability is required (and only outsider confidentiality), this approach still yield the lowest ciphertext overhead (roughly 25% shorter than the scheme by Zheng [47]), but is not as computationally efficient as [47]. If insider confidentiality is required (and only outsider unforgeability), this approach yields a scheme with exactly the same ciphertext overhead and slightly more expensive computational cost than the currently most efficient scheme by Gorantla *et al.* [20] instantiated with HMQV [27]. However, our approach is secure in a stronger security model.

We furthermore propose a simple optimization of the "sign-then-encrypt" and "encrypt-then-sign" constructions of signcryption, using tag-based encryption (TBE) [34,25][2]. While both constructions are shown to be insider secure, the latter requires a special one-to-one property of the signature scheme which, in practice, limits instantiations to the random oracle model. However, the advantage of this approach is that it achieves strong unforgeability which is not achieved by the former approach. To instantiate these schemes, we show how the most efficient standard and random oracle model PKE schemes can be turned into TBE schemes with practically no additional cost. This leads to an insider secure random oracle model scheme that is at least as efficient as any other existing scheme both in terms of ciphertext overhead and computational cost, as well as efficient standard model schemes.

Lastly, we show how a signature scheme and an encryption scheme which satisfy a few special requirements can be combined in a non-black-box way to achieve higher efficiency than a simple composition. The basic idea of this approach is simple and essentially lets the signature and encryption use "shared randomness". We call schemes that can be combined in this way *signcryption-composable*, and we show that some of the most efficient standard model encryption and signature schemes satisfy this. The resulting signcryption schemes are the most efficient insider secure standard model schemes.

We emphasize that the advantage of the above compositions lies not only in the achieved efficiency by the obtained signcryption schemes, but also in their simplicity, which allows us to prove security using already established security results for the underlying primitives. We believe that the constructions obtained via our compositions can be used as a benchmark to evaluate future signcryption schemes.

While in this paper we concentrate on schemes providing the basic security properties of signcryption, i.e. confidentiality and unforgeability, we conjecture that schemes providing additional properties, such as non-repudiation and anonymity, can be constructed using similar techniques.

[2] TBE has previously been introduced under the name *encryption with labels* [37].

2 Building Blocks

In our constructions of signcryption schemes we will make use of a number of different primitives including tag-based encryption (TBE), tag-based key encapsulation mechanism (TBKEM), signatures, symmetric key encryption (SKE), data encapsulation mechanism (DEM), message authentication codes (MAC), and tag-based non-interactive key establishment (TNIKE).

A TBE scheme is a public key encryption scheme in which the encryption and decryption algorithm take a tag as an additional input, and has been used in several other papers (e.g [37,34,25]). We will use TBE schemes which provide full CCA security [25] and a weaker selective tag variant, which we will denote `IND-tag-CCA` and `IND-stag-CCA`, respectively. A TBKEM[3] is the key encapsulation analogue of a TBE scheme for which we will also consider the security notions `IND-tag-CCA` and `IND-stag-CCA`.

For signatures, we use the standard security definitions of weak and strong unforgeability [3], denoted `wUF-CMA` and `sUF-CMA`, for SKE we use the security notions `IND-CPA`, `IND-CCA` and `INT-CTXT` as defined in [7], and for MAC we use the security notions `wUF-CMA` and `sUF-CMA` [7]. We define a DEM to be a special case of a SKE in which the encryption algorithm is deterministic.

A non-interactive key exchange (NIKE), introduced in [19] and formally defined in [15], is given by a setup algorithm `Setup` which returns a set of public parameters par, a key generation algorithm `KG` which on input par returns a public/private key pair (pk, sk), a shared key generation algorithm `Share` which on input par, a public key of one entity pk_1 and a private key of another entity sk_2, returns a shared key K. It is required for all $par \leftarrow \mathtt{Setup}(1^k)$ and all (pk_1, sk_1) and (pk_2, sk_2) output from $\mathtt{KG}(par)$ that $\mathtt{Share}(par, pk_1, sk_2) = \mathtt{Share}(par, pk_2, sk_1)$. A TNIKE is a tag-based extension of a NIKE in which the shared key generation algorithm takes as additional input a tag. We require a (T)NIKE to be secure against *active attacks* [15].

Due to space limitations, the formal definitions of these primitives are not included, and we refer the reader to the full version of the paper [35] for these.

3 Signcryption

A signcryption scheme is given by the following algorithms: a setup algorithm `Setup` which on input 1^k returns a set of public parameters par; a sender key generation algorithm $\mathtt{KG}_S$ which on input par returns a public/private sender key pair (pk_S, sk_S); a receiver key generation algorithm $\mathtt{KG}_R$ which on input par returns a public/private receiver key pair (pk_R, sk_R); a signcryption algorithm `SC` which on input par, sk_S, pk_R, and a message m, returns a ciphertext c; an unsigncryption algorithm `USC` which on input par, pk_S, sk_R, and c, returns either m or an error symbol $\perp$.

[3] Note that this primitive is *different* from the *Tag-KEM* introduced in [2], although they are closely related. It is easy to see that every `IND-CCA` Tag-KEM can be used as an `IND-tag-CCA` TBKEM.

It is required for all $par \leftarrow \mathtt{Setup}(1^k)$, all $(pk_S, sk_S) \leftarrow \mathtt{KG}_S(par)$, all $(pk_R, sk_R) \leftarrow \mathtt{KG}_R(par)$, and all messages m that $m = \mathtt{USC}(par, pk_S, sk_R, \mathtt{SC}(par, sk_S, pk_R, m))$.

3.1 Security

As mentioned in the introduction, a multi-user security definition is required for signcryption schemes. However, a number of slightly different models have been introduced in the literature (e.g. see [3,6,31,20]). In the following definitions, the differences of these will be highlighted. We firstly consider security models with insider security, and then discuss the weaker outsider counterparts.

Confidentiality. The strongest notion of confidentiality was introduced in [31] and is based on a security model in which the adversary can freely choose all user keys, except the challenge receiver key. We refer to this model as the *dynamic multi-user* model, and in this model we consider the notion *indistinguishability against insider chosen ciphertext attacks* (dM-IND-iCCA). More specifically, for a signcryption scheme $SC = (\mathtt{Setup}, \mathtt{KG}_S, \mathtt{KG}_R, \mathtt{SC}, \mathtt{USC})$ and a security parameter 1^k, dM-IND-iCCA security is defined via the experiment $\mathtt{Exp}^{\mathtt{dM\text{-}IND\text{-}iCCA}}_{SC,\mathcal{A}}(k)$ shown in Fig. 1 (upper left). In the experiment, the adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ has access to an unsigncryption oracle $\mathcal{O} = \{\mathtt{Unsigncrypt}\}$ which is defined as follows:

- `Unsigncrypt:` Given a public sender key pk_S and ciphertext c, the oracle returns $m/\perp \leftarrow \mathtt{USC}(par, pk_S, sk^*_R, c)$ where sk^*_R is the private receiver key generated in the beginning of the experiment. A query of the form (pk^*_S, c^*), where pk^*_S is the challenge sender key specified by $\mathcal{A}$ and c^* is the challenge ciphertext, is not allowed.

A security model defining a slightly weaker security notion was used in [3,5]. In this security model, which we will refer to as the *fixed challenge key multi-user* model, the adversary cannot choose the challenge sender key. More specifically, in this model we define indistinguishability against insider chosen ciphertext attacks (fM-IND-iCCA) for a signcryption scheme SC and security parameter 1^k via the experiment $\mathtt{Exp}^{\mathtt{fM\text{-}IND\text{-}iCCA}}_{SC,\mathcal{A}}(k)$ shown in Fig. 1 (upper right). In the experiment, $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ has access to an unsigncryption oracle as defined above.

Definition 1. *A signcryption scheme SC is said to be* X-IND-iCCA *secure, if* $|\Pr[\mathtt{Exp}^{\mathtt{X\text{-}IND\text{-}iCCA}}_{SC,\mathcal{A}}(k) = 1] - 1/2|$ *is negligible in k for any probabilistic polynomial-time algorithm $\mathcal{A}$, where* $\mathtt{X} \in \{\mathtt{dM}, \mathtt{fM}\}$.

Unforgeability. Like the confidentiality definition above, we consider unforgeability in both the dynamic and the fixed challenge key multi-user models. For a signcryption scheme SC and security parameter 1^k, we define (weak) unforgeability against insider chosen message attacks in the dynamic multi-user model (dM-wUF-iCMA) via experiment $\mathtt{Exp}^{\mathtt{dM\text{-}wUF\text{-}iCMA}}_{SC,\mathcal{A}}(k)$ shown in Fig. 1 (lower left). In the experiment, the adversary $\mathcal{A}$ has access to a signcryption oracle $\mathcal{O} = \{\mathtt{Signcrypt}\}$ defined as follows:

$\mathbf{Exp}^{\mathtt{dM\text{-}IND\text{-}iCCA}}_{SC,\mathcal{A}}(k)$:
$par \leftarrow \mathtt{Setup}(1^k)$
$(pk^*_R, sk^*_R) \leftarrow \mathtt{KG}_R(par)$
$(pk^*_S, sk^*_S, m_0, m_1, st) \leftarrow \mathcal{A}_1^{\mathcal{O}}(par, pk^*_R)$
$b \leftarrow \{0,1\}$; $c^* \leftarrow \mathtt{SC}(par, sk^*_S, pk^*_R, m_b)$
$b' \leftarrow \mathcal{A}_2^{\mathcal{O}}(st, c^*)$
If $b = b'$ return 1
Else return 0

$\mathbf{Exp}^{\mathtt{fM\text{-}IND\text{-}iCCA}}_{SC,\mathcal{A}}(k)$:
$par \leftarrow \mathtt{Setup}(1^k)$
$(pk^*_S, sk^*_S) \leftarrow \mathtt{KG}_S(par)$
$(pk^*_R, sk^*_R) \leftarrow \mathtt{KG}_R(par)$
$(m_0, m_1, st) \leftarrow \mathcal{A}_1^{\mathcal{O}}(par, pk^*_S, sk^*_S, pk^*_R,)$
$b \leftarrow \{0,1\}$; $c^* \leftarrow \mathtt{SC}(par, sk^*_S, pk^*_R, m_b)$
$b' \leftarrow \mathcal{A}_2^{\mathcal{O}}(st, c^*)$
If $b = b'$ return 1
Else return 0

$\mathbf{Exp}^{\mathtt{dM\text{-}wUF\text{-}iCMA}}_{SC,\mathcal{A}}(k)$:
$L \leftarrow \emptyset$; $par \leftarrow \mathtt{Setup}(1^k)$
$(pk^*_S, sk^*_S) \leftarrow \mathtt{KG}_S(par)$
$(pk^*_R, sk^*_R, c^*) \leftarrow \mathcal{A}^{\mathcal{O}}(par, pk^*_S)$
$m^* \leftarrow \mathtt{USC}(par, pk^*_S, sk^*_R, c^*)$
If $m^* \neq \perp \wedge (m^*, pk^*_R) \notin L$ return 1
Else return 0

$\mathbf{Exp}^{\mathtt{fM\text{-}wUF\text{-}iCMA}}_{SC,\mathcal{A}}(k)$:
$L \leftarrow \emptyset$; $par \leftarrow \mathtt{Setup}(1^k)$
$(pk^*_S, sk^*_S) \leftarrow \mathtt{KG}_S(par)$;
$(pk^*_R, sk^*_R) \leftarrow \mathtt{KG}_R(par)$
$c^* \leftarrow \mathcal{A}^{\mathcal{O}}(par, pk^*_S, pk^*_R, sk^*_R)$
$m^* \leftarrow \mathtt{USC}(par, pk^*_S, sk^*_R, c^*)$
If $m^* \neq \perp \wedge (m^*, pk^*_R) \notin L$ return 1
Else return 0

Fig. 1. Experiments for confidentiality and unforgeability

– `Signcrypt`: Given a public receiver key pk_R and a message m, the oracle returns $c \leftarrow \mathtt{SC}(par, sk^*_S, pk_R, m)$, where sk^*_S is the secret sender key generated in the beginning of the experiment. Furthermore, (pk_R, m) is added to the list L.

Likewise, we define (weak) unforgeability against insider chosen message attacks in the fixed challenge key multi-user model (`fM-wUF-iCMA`) via experiment $\mathbf{Exp}^{\mathtt{fM\text{-}UF\text{-}iCMA}}_{SC,\mathcal{A}}(k)$ shown in Fig. 1 (lower right), where $\mathcal{A}$ has access to a signcryption oracle as defined above.

Definition 2. *A signcryption scheme SC is said to be* `X-wUF-iCMA` *secure, if* $\Pr[\mathbf{Exp}^{\mathtt{X\text{-}wUF\text{-}iCMA}}_{SC,\mathcal{A}}(k) = 1]$ *is negligible in k for any probabilistic polynomial-time algorithm* $\mathcal{A}$*, where* $\mathtt{X} \in \{\mathtt{dM}, \mathtt{fM}\}$.

Strong insider unforgeability (`dM-sUF-iCMA` and `fM-sUF-iCMA` security) is defined in a similar way to the above, with the only change that the list L now contains (pk_R, m, c) for signcryptions queries made by $\mathcal{A}$, and it is required that $(pk^*_R, m^*, c^*) \notin L$ for the forgery output by $\mathcal{A}$.

Note that Libert *et al.* [32,30] uses a different unforgeability definition which is concerned about signature extracted from a ciphertext. However, this does not imply the unforgeability mentioned here. (In fact, the scheme proposed in [30] is insecure according to the above definition.)

Outsider Security. While insider security is inherent in the dynamic multi-user model, we can consider a weaker version of the fixed challenge key multi-user model in which the adversary knows neither the private sender key nor the private receiver key for the challenge key pairs. This is modeled by limiting the input given to the adversary $\mathcal{A}$ to (par, pk^*_S, pk^*_R) in the experiments $\mathbf{Exp}^{\mathtt{fM\text{-}IND\text{-}iCCA}}_{SC,\mathcal{A}}(k)$

and $\mathtt{Exp}^{\mathtt{fM\text{-}wUF\text{-}iCMA}}_{SC,\mathcal{A}}(k)$ defined above. However, with this limited input, $\mathcal{A}$ can no longer compute signcryptions using the challenge private sender key sk^*_S in $\mathtt{Exp}^{\mathtt{fM\text{-}IND\text{-}iCCA}}_{SC,\mathcal{A}}(k)$, and can no longer compute unsigncryptions using the challenge private receiver key sk^*_R in $\mathtt{Exp}^{\mathtt{fM\text{-}wUF\text{-}iCMA}}_{SC,\mathcal{A}}(k)$. Hence, in both experiments, $\mathcal{A}$ is given access to oracles $\mathcal{O} = \{\mathtt{Signcrypt}, \mathtt{Unsigncrypt}\}$ defined as in the above.

We denote these modified experiments $\mathtt{Exp}^{\mathtt{fM\text{-}IND\text{-}oCCA}}_{SC,\mathcal{A}}(k)$ and $\mathtt{Exp}^{\mathtt{fM\text{-}wUF\text{-}oCMA}}_{SC,\mathcal{A}}(k)$, and define the outsider security notions `fM-IND-oCCA` and `fM-wUF-oCMA` in a similar way to the corresponding insider security notions. We furthermore consider the strong variant of the unforgeability notion `fM-wUF-oCMA` which will be denoted `fM-sUF-oCMA`, and is defined in a similar way to the corresponding insider notion `fM-sUF-iCMA`.

Key Registration. In the above experiments, the adversary can freely choose the public keys submitted to signcryption and unsigncryption oracles. However, in systems based on a traditional PKI, users are required to obtain a certificate by registering their public key at a certificate authority before the public key can be used in interaction with other users. This allows additional security measures such as requiring that a user prove knowledge of the secret key corresponding to the public key he is registering. To model security in this scenario, we give the adversary access to a *key registration oracle* in addition to normal queries. The key registration oracle maintains a list L_{PK} of registered key pairs and interacts with $\mathcal{A}$ as follows:

`Register-key:` Given a key pair (pk, sk), the oracle checks if (pk, sk) is a valid key pair. If not, the oracle returns 0. Otherwise, it adds (pk, sk) to L_{PK}, and returns 1.

When $\mathcal{A}$ submits a signcryption query (pk_R, m) or an unsigncryption query (pk_S, c), it is then required that $(pk_R, *) \in L_{PK}$ and $(pk_S, *) \in L_{PK}$, respectively. We write, for example, `dM-sUF-iCMA(KR)` to mean `dM-sUF-iCMA` security with key registration in order to distinguish it from ordinary `dM-sUF-iCMA`.

Key registration has been used in connection with the dynamic multi-user model in [44]. Furthermore, Gorantla *et al.* [20] defines a multi-user security model in which the adversary cannot choose any of the keys used in the system, but is only given a list of public user keys and access to a corruption oracle. This model implicitly implies key registration and we refer to this as the *static multi-user* model (see [20] for the details of the security definitions in this model). Furthermore, we use the prefix `sM-` to denote this model. We note that dynamic and fixed challenge key multi-user security with key registration trivially implies the static multi-user security.

Comparison of Security Notions. The hierarchy of the above mentioned security notions is shown in Fig. 2. The proofs of the implications shown in the figure are straightforward and are not given here. We furthermore conjecture that a separation exists between all of the security notions shown in the figure.

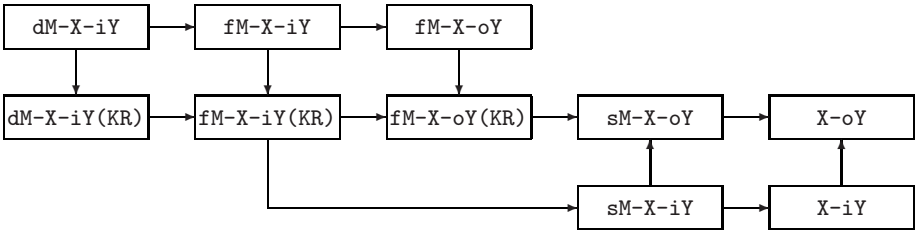

Fig. 2. Implications between security notions. In the figure, "A → B" means that security wrt. security notion A implies security wrt. security notion B. (X, Y) is (IND, CCA) for confidentiality, and is either (wUF, CMA) or (sUF, CMA) for unforgeability. The security notions without any prefix {dM-, fM-, sM-} indicate the two-user security notion.

$\mathtt{Setup}_{sc}(1^k)$: $par_n \leftarrow \mathtt{Setup}_n(1^k)$ $par_{te} \leftarrow \mathtt{Setup}_{te}(1^k)$ Output $par \leftarrow (par_n, par_{te})$.	$\mathtt{SC}(par, sk_S, pk_R, m)$: $\mathsf{tag} \leftarrow pk_S$ $c_E \leftarrow \mathtt{TEnc}(par_{te}, pk_{R2}, \mathsf{tag}, m)$ $K \leftarrow \mathtt{Share}(par_n, pk_{R1}, sk_S)$ $\sigma \leftarrow \mathtt{Mac}(K, (pk_{R2}\|c_E))$ Output $c \leftarrow (c_E, \sigma)$.
$\mathtt{KG}_S(par)$: Output $(pk_S, sk_S) \leftarrow \mathtt{KG}_n(par_n)$.	
$\mathtt{KG}_R(par)$: $(pk_{R1}, sk_{R1}) \leftarrow \mathtt{KG}_n(par_n)$ $(pk_{R2}, sk_{R2}) \leftarrow \mathtt{KG}_{te}(par_{te})$ $pk_R \leftarrow (pk_{R1}, pk_{R2})$ $sk_R \leftarrow (sk_{R1}, sk_{R2})$ Output (pk_R, sk_R).	$\mathtt{USC}(par, pk_S, sk_R, c)$: Parse c as (c_E, σ). $\mathsf{tag} \leftarrow pk_S$ $K \leftarrow \mathtt{Share}(par_n, pk_S, sk_{R1})$ If $\mathtt{MVer}(K, (pk_{R2}\|c_E), \sigma) = \perp$ then output $\perp$ and stop. Output $\mathtt{TDec}(par_{te}, sk_{R2}, \mathsf{tag}, c_E)$.

Fig. 3. Simple composition using symmetric key primitives: TEtK&M

4 Simple Composition Using Symmetric Key Primitives

In this section we show that if only outsider security is required for either confidentiality or unforgeability (or for both), then symmetric key primitives can be used to construct efficient signcryption schemes. However, in order to make use of symmetric key primitives, sender and receiver must share a symmetric key. To achieve this, we employ a (T)NIKE which has the advantage of not requiring the sender and receiver to exchange messages to compute a shared key. As we will see, the combination of symmetric key primitives and (T)NIKE schemes secure against active attacks provides the strongest notion of outsider security. These constructions are only interesting if efficient instantiations of (T)NIKE schemes secure against active attacks can be constructed. However, in Section 7 we show that this is indeed possible. Due to space limitations, all proofs of the theorems in this section are given in the full version [35].

*Tag-based-Encrypt then Key-exchange and MAC (*TEtK&M*).* Firstly, we consider a construction in which outsider unforgeability is achieved by the combined use of a NIKE and a MAC scheme, and which we call *"Tag-based-Encrypt then*

$\mathtt{Setup}_{sc}(1^k)$: $par_{tn} \leftarrow \mathtt{Setup}_{tn}(1^k)$ $par_{sig} \leftarrow \mathtt{Setup}_{sig}(1^k)$ Output $par \leftarrow (par_{tn}, par_{sig})$.	$\mathtt{SC}(par, sk_S, pk_R, m)$: $\mathtt{tag} \leftarrow pk_{S2}$ $K \leftarrow \mathtt{TShare}(par_{tn}, pk_R, sk_{S1}, \mathtt{tag})$ $c_E \leftarrow \mathtt{SEnc}(K, m)$ $\sigma \leftarrow \mathtt{Sign}(par_{sig}, sk_{S2}, (pk_R \| c_E))$ Output $c \leftarrow (c_E, \sigma)$.
$\mathtt{KG}_S(par)$: $(pk_{S1}, sk_{S1}) \leftarrow \mathtt{KG}_{tn}(par_{tn})$ $(pk_{S2}, sk_{S2}) \leftarrow \mathtt{KG}_{sig}(par_{sig})$ $pk_S \leftarrow (pk_{S1}, pk_{S2})$ $sk_S \leftarrow (sk_{S1}, sk_{S2})$ Output (pk_S, sk_S).	$\mathtt{USC}(par, pk_S, sk_R, c)$: Parse c as (c_E, σ), $\mathtt{tag} \leftarrow pk_{S2}$ If $\mathtt{SVer}(par_{sig}, pk_{S2}, (pk_R \| c_E), \sigma) = \perp$ then output $\perp$ and stop. $K \leftarrow \mathtt{TShare}(par_{tn}, pk_{S1}, sk_R, \mathtt{tag})$ Output $\mathtt{SDec}(K, c_E)$.
$\mathtt{KG}_R(par)$: Output $(pk_R, sk_R) \leftarrow \mathtt{KG}_{tn}(par_{tn})$.	

Fig. 4. Simple composition using symmetric key primitives: TK&SEtS

$\mathtt{Setup}_{sc}(1^k)$: Output $par \leftarrow \mathtt{Setup}_n(1^k)$.	$\mathtt{SC}(par, sk_S, pk_R, m)$: $K \leftarrow \mathtt{Share}(par, pk_R, sk_S)$ Output $c \leftarrow \mathtt{SEnc}(K, m)$.
$\mathtt{KG}_S(par)$: Output $(pk_S, sk_S) \leftarrow \mathtt{KG}_n(par)$.	$\mathtt{USC}(par, pk_S, sk_R, c)$: $K \leftarrow \mathtt{Share}(par, pk_S, sk_R)$ Output $\mathtt{SDec}(K, c)$.
$\mathtt{KG}_R(par)$: Output $(pk_R, sk_R) \leftarrow \mathtt{KG}_n(par)$.	

Fig. 5. Simple composition using symmetric key primitives: K&SE

Key-exchange and MAC" (TEtK&M). More specifically, let $N = (\mathtt{Setup}_n, \mathtt{KG}_n, \mathtt{Share})$ be a NIKE scheme, let $TE = (\mathtt{Setup}_{te}, \mathtt{KG}_{te}, \mathtt{TEnc}, \mathtt{TDec})$ be a TBE scheme, and let $M = (\mathtt{Mac}, \mathtt{MVer})$ be a MAC scheme. Then TEtK&M is defined as shown in Fig. 3. The security of the scheme is provided by the following theorems. Note that the MAC scheme M is required to be *one-to-one*[4] to guarantee confidentiality.

Theorem 3. *Assume TE is* IND-tag-CCA *(resp.* IND-stag-CCA*) secure and M is one-to-one. Then* TEtK&M *is* dM-IND-iCCA *(resp.* fM-IND-iCCA*) secure.*

Theorem 4. *Assume N is secure against active attacks and M is* sUF-CMA *(resp.* wUF-CMA*) secure. Then* TEtK&M *is* fM-sUF-oCMA *(resp.* fM-wUF-oCMA*) secure.*

*Tag-based-Key-exchange and Symmetric-key-Encrypt then Sign (*TK&SEtS*).* Using a similar approach to the above, we consider a signcryption scheme in which outsider confidentiality is achieved by the combined use of a TNIKE scheme and a SKE scheme, and which we call *"Tag-based-Key-exchange and Symmetric-key-Encrypt then Sign"* (TK&SEtS). For this scheme, the tag-based property of TNIKE is required to ensure that a ciphertext is only valid under a single public sender key. Specifically, let $TN = (\mathtt{Setup}_{tn}, \mathtt{KG}_{tn}, \mathtt{TShare})$ be a TNIKE

[4] A MAC is said to be one-to-one if given a key K and a message m, there is only one MAC tag σ such that $\mathtt{MVer}(K, m, \sigma) = \top$.

scheme, let $S = (\mathtt{Setup}_{sig}, \mathtt{KG}_{sig}, \mathtt{Sign}, \mathtt{SVer})$ be a signature scheme, and let $SE = (\mathtt{SEnc}, \mathtt{SDec})$ be a SKE scheme. Then TK&SEtS is defined as shown in Fig. 4. The security of this scheme is provided by the following theorems. Note that the SKE scheme SE is only required to be IND-CPA secure to guarantee confidentiality.

Theorem 5. *Assume S is* sUF-CMA *secure, TN is secure against active attacks, and SE is* IND-CPA *secure. Then* TK&SEtS *is* fM-IND-oCCA *secure.*

Theorem 6. *Assume S is* sUF-CMA *(resp.* wUF-CMA*) secure. Then* TK&SEtS *is* dM-sUF-iCMA *(resp.* dM-wUF-iCMA*) secure.*

*Key-exchange then Symmetric-key-Encrypt (*K&SE*).* Finally, we consider a signcryption scheme providing outsider unforgeability and outsider confidentiality. This scheme, which we call *"Key-exchange and Symmetric-key-Encrypt"* (K&SE), consists only of a NIKE scheme and a SKE scheme satisfying the security of *authenticated encryption* [7]. Interestingly, in this scheme a ciphertext consists only of the output of the underlying SKE scheme. Specifically, let $N = (\mathtt{Setup}_n, \mathtt{KG}_n, \mathtt{Share})$ be a NIKE scheme, and let $SE = (\mathtt{SEnc}, \mathtt{SDec})$ be a SKE scheme. Then K&SE is defined as shown in Fig. 5. The following state that K&SE satisfies both outsider confidentiality and outsider unforgeability.

Theorem 7. *Assume N is secure against active attacks and SE is* IND-CCA *secure. Then* K&SE *is* fM-IND-oCCA *secure.*

Theorem 8. *Assume N is secure against active attacks and SE is* INT-CTXT *secure. Then* K&SE *is* fM-sUF-oCMA *secure.*

5 Simple Composition Using Tag-Based Encryption

An *et al.* [3] analyzed the security of the simple composition of signature and encryption, and showed that both sign-then-encrypt and encrypt-then-sign are secure, but only for a weaker notion of confidentiality termed *generalized IND-CCA* security. If ordinary IND-CCA security is required, the latter becomes insecure, even if the used signature scheme is strongly unforgeable and the encryption scheme is IND-CCA secure. Furthermore, simple composition does not yield multi-user security. In [3], this is overcome by including the public sender key in the plaintext, and the public receiver key in the input to the signing algorithm. While this achieves multi-user security, it also introduces additional ciphertext overhead.

Here we show that by using a TBE scheme, multi-user security can be achieved without introducing additional ciphertext overhead. This is of course only useful if it is possible to construct TBE schemes which do not have a higher ciphertext overhead than ordinary PKE schemes. In Section 7 we show that this is indeed possible for the currently most efficient encryption schemes in both the standard and the random oracle model. Due to space limitations, the proofs of the theorems in this section are given in the full version [35].

$\mathtt{Setup}_{sc}(1^k)$:	$\mathtt{KG}_S(par)$:
$par_{te} \leftarrow \mathtt{Setup}_{te}(1^k)$	Output $(pk_S, sk_S) \leftarrow \mathtt{KG}_{sig}(par_{sig})$.
$par_{sig} \leftarrow \mathtt{Setup}_{sig}(1^k)$	$\mathtt{KG}_R(par)$:
Output $par \leftarrow (par_{te}, par_{sig})$.	Output $(pk_R, sk_R) \leftarrow \mathtt{KG}_{te}(par_{te})$.

Sign-then-Tag-based-Encrypt StTE	
$\mathtt{SC}(par, sk_S, pk_R, m)$:	$\mathtt{USC}(par, pk_S, sk_R, c)$:
$\sigma \leftarrow \mathtt{Sign}(par_{sig}, sk_S, (pk_R \| m))$	$\mathtt{tag} \leftarrow pk_S$
$\mathtt{tag} \leftarrow pk_S$	$(m \| \sigma)/\perp \leftarrow \mathtt{TDec}(par_{te}, sk_R, \mathtt{tag}, c)$
$c \leftarrow \mathtt{TEnc}(par_{te}, pk_R, \mathtt{tag}, (m \| \sigma))$	(if output is $\perp$, then output $\perp$ and stop.)
Output c.	If $\mathtt{SVer}(par_{sig}, pk_S, (pk_R \| m), \sigma) = \top$
	then output m, otherwise output $\perp$.

Tag-based-Encrypt-then-Sign TEtS	
$\mathtt{SC}(par, sk_S, pk_R, m)$:	$\mathtt{USC}(par, pk_S, sk_R, c)$:
$\mathtt{tag} \leftarrow pk_S$	Parse c as (c_E, σ), $\mathtt{tag} \leftarrow pk_S$
$c_E \leftarrow \mathtt{TEnc}(par_{te}, pk_R, \mathtt{tag}, m)$	If $\mathtt{SVer}(par_{sig}, pk_S, (pk_R \| c_E), \sigma) = \perp$
$\sigma \leftarrow \mathtt{Sign}(par_{sig}, sk_S, (pk_R \| c_E))$	then output $\perp$ and stop.
Output $c \leftarrow (c_E, \sigma)$.	Output $\mathtt{TDec}(par_{te}, sk_R, \mathtt{tag}, c_E)$.

Fig. 6. Simple composition of signature and TBE. Note that StTE and TEtS use the same setup and key generation algorithms.

Let $TE = (\mathtt{Setup}_{te}, \mathtt{KG}_{te}, \mathtt{TEnc}, \mathtt{TDec})$ be a TBE scheme and let $S = (\mathtt{Setup}_{sig}, \mathtt{KG}_{sig}, \mathtt{Sign}, \mathtt{SVer})$ be a signature scheme. Then the *"Sign-then-Tag-based-Encrypt"* (StTE) and *"Tag-based-Encrypt-then-Sign"* (TEtS) schemes are defined as shown in Fig. 6. We achieve the following security results for StTE.

Theorem 9. *Assume TE is* IND-tag-CCA *(resp.* IND-stag-CCA*) secure. Then* StTE *is* dM-IND-iCCA *(resp.* fM-IND-iCCA*) secure.*

Theorem 10. *Assume S is* wUF-CMA *secure. Then* StTE *is* dM-wUF-iCMA *secure.*

Note that the receiver trivially obtains a publicly verifiable signature of the sender on the sent message m when unsigncrypting a valid ciphertext. Hence, the receiver can convince any third party that the message m was indeed sent by the sender (this provides a similar type of non-repudiation to [32], which introduces the notion of *detachable signatures*).

Like the encrypt-then-sign approach, TEtS will generally not achieve IND-CCA security, even if the used TBE scheme is IND-CCA secure. However, if the signature scheme is *one-to-one*[5] the following results can be obtained.

Theorem 11. *Assume TE is* IND-tag-CCA *(resp.* IND-stag-CCA*) secure and S is one-to-one. Then* TEtS *is* dM-IND-iCCA *(resp.* fM-IND-iCCA*) secure.*

Furthermore, unlike StTE, if a strongly unforgeable signature scheme is used, TEtS will also be strongly unforgeable (note that the one-to-one property is not required in the following theorem).

[5] A signature scheme is said to be one-to-one if given public parameters par, a public key pk, and a message m, there exists only *one* signature σ such that $\mathtt{SVer}(par, pk, m, \sigma) = \top$.

Theorem 12. *Assume S is* sUF-CMA *(resp.* wUF-CMA*) secure. Then* TEtS *is* dM-sUF-iCMA *(resp.* dM-wUF-iCMA*) secure.*

Currently, only random oracle model signature schemes, like BLS [10], have the one-to-one property. However, BLS is one of the most efficient schemes in terms of signature size and signing cost, and as we will see in Section 9, constructing TEtS using BLS and a tag-based variant of DHIES [1] (see also Section 7) will yield an insider secure signcryption scheme in the random oracle model, which is at least as efficient as the currently most efficient insider secure schemes by Libert *et al.* [29] and by Li *et al.* [28] which are also inspired by BLS and the DHIES scheme.

6 Signcryption Composability

While the simple composition of signature and TBE yields signcryption schemes which are at least as efficient as any other insider secure signcryption scheme (see Section 9), a part of the original motivation for considering signcryption as a separate primitive, is to achieve higher efficiency than such black-box compositions. In this section we show how to achieve insider secure signcryption schemes in the standard model which is more efficient than a black-box composition of the most efficient standard model signature and encryption schemes.

The idea behind our approach is fairly simple. Since both signature and encryption in the standard model are probabilistic, the sender could potentially reuse the same "randomness" for both signing and encryption. By doing so, both ciphertext and computational overhead can potentially be reduced. Naturally, a signature and an encryption schemes need to "match" to enable this, and to be able to prove security of the resulting signcryption scheme, we require the individual schemes to have a few special properties. We say that a pair of schemes satisfying these requirements are *signcryption composable* (SC-composable), and we will formally define the requirements below. Since we adopt the KEM/DEM approach, our SC-definition will be concerned with a signature scheme and a TBKEM scheme. We furthermore assume that both the TBKEM and the signature scheme are *patitionable*[6] i.e. for a TBKEM, it is required that the encapsulation algorithm can be divided into two deterministic algorithms TE_1 and TE_2 such that an encapsulation of a key can be computed by picking random $r \leftarrow \mathcal{R}$ (the randomness space $\mathcal{R}$ is specified by par), computing $c_1 \leftarrow \mathsf{TE}_1(par, r)$ and $(c_2, K) \leftarrow \mathsf{TE}_2(par, pk, \mathsf{tag}, r)$, and returning the encapsulation $c \leftarrow (c_1, c_2)$ and the encapsulated key K, and given c_1 and a tag, there is at most one c_2 and one K such that $\mathtt{TDecap}(par, sk, \mathsf{tag}, (c_1, c_2)) = K$. Partitionability of a signature scheme is defined in a similar way, and we let S_1 and S_2 denote the message independent part (taking only par and r as input) and the message dependent part of the signing algorithm, respectively.

Definition 13. *We say that a partitionable TBKEM* $TK = (\mathtt{Setup}_{tk}, \mathtt{KG}_{tk}, \mathtt{TEncap}, \mathtt{TDecap})$ *and a partitionalble signature scheme* $S = (\mathtt{Setup}_{sig}, \mathtt{KG}_{sig},$

[6] Partitionability of a signature scheme has previously been defined in [11].

$\mathtt{Sign}, \mathtt{SVer}$) *are* signcryption composable *(SC-composable) if they satisfy the following:*

- ***Property 1.*** *(Compatible Setup) There exists an algorithm* $\mathtt{Setup}'_{tk}$ *that, given public parameters* $par_{sig} \leftarrow \mathtt{Setup}_{sig}(1^k)$ *as input, generates* par_{tk} *distributed identically to the output of* $\mathtt{Setup}_{tk}(1^k)$*. Furthermore, there exists an algorithm* $\mathtt{Setup}'_{sig}$ *that, given* $par_{tk} \leftarrow \mathtt{Setup}_{tk}(1^k)$ *as input, generates* par_{sig} *distributed identically to the output of* $\mathtt{Setup}_{sig}(1^k)$*.*
- ***Property 2.*** *(Shared Randomness) Let* $\mathcal{R}_{tk}$ *and* $\mathcal{R}_{sig}$ *be the randomness spaces specified by* par_{tk} *and* par_{sig} *used by* $\mathtt{TEncap}$ *of* TK *and* $\mathtt{Sign}$ *of* S*, respectively. It is required that*
  - $\mathcal{R}_{sig} = \mathcal{R}_{tk} \times \mathcal{R}^+_{sig}$ *i.e. the randomness space for* TK *is shared by both* TK *and* S *(in the following we will use* $\mathcal{R}$ *to denote the common randomness space). We allow* $\mathcal{R}^+_{sig}$ *to be empty.*
  - *For all choices of* $(r, s) \in \mathcal{R} \times \mathcal{R}^+_{sig}$ *and all* $\sigma_1 \leftarrow \mathsf{S}_1(par_{sig}, (r, s))$*, it is required that* σ_1 *can be written as* $\sigma_1 = (c_1, \sigma'_1)$ *such that* $c_1 = \mathsf{TE}_1(par_{tk}, r)$*. We allow* σ'_1 *to be an empty string.*
- ***Property 3.*** *(Signature/Ciphertext Simulatability) There exist algorithms* S'_1*,* S'_2 *and* TE'_2 *with the following properties:*
  - TE'_2*: Given* par_{tk}*, a public/private key pair* $(pk_{tk}, sk_{tk}) \leftarrow \mathtt{KG}_{tk}(par_{tk})$*, a tag* tag*, and* $c_1 = \mathsf{TE}_1(par_{tk}, r)$ *for some* $r \in \mathcal{R}$*, this algorithm outputs* c_2 *and* K *such that* $(c_2, K) = \mathsf{TE}_2(par_{tk}, pk_{tk}, \mathsf{tag}, r)$*.*
  - S'_1*: Given* par_{sig}*,* $c_1 = \mathsf{TE}_1(par_{tk}, r)$ *for some* $r \in \mathcal{R}$*, and* $s \in \mathcal{R}^+_{sig}$*, this algorithm outputs* σ'_1 *such that* $(c_1, \sigma'_1) = \mathsf{S}_1(par_{sig}, (r, s))$*. If* $\mathcal{R}^+_{sig}$ *is empty, we do not consider this algorithm.*
  - S'_2*: Given* par_{sig}*,* $(pk_{sig}, sk_{sig}) \leftarrow \mathtt{KG}_{sig}(par_{sig})$*, a message* m*,* $c_1 = \mathsf{TE}_1(par_{tk}, r)$ *for some* $r \in \mathcal{R}$*, and* $s \in \mathcal{R}^+_{sig}$*, this algorithm outputs* σ_2 *such that* $\sigma_2 = \mathsf{S}_2(par_{sig}, sk_{sig}, m, (r, s))$*.*

Although the requirements might seem somewhat restrictive, as shown in Section 8, tag-based variants of many of the existing standard model KEMs are in fact SC-composable with a number of standard model signature schemes.

6.1 Signcryption from SC-Composable Schemes

Let $TK = (\mathtt{Setup}_{tk}, \mathtt{KG}_{tk}, \mathtt{TEncap}, \mathtt{TDecap})$ be a partitionable TBKEM scheme in which $\mathtt{TEncap} = (\mathsf{TE}_1, \mathsf{TE}_2)$, let $S = (\mathtt{Setup}_{sig}, \mathtt{KG}_{sig}, \mathtt{Sign}, \mathtt{SVer})$ be a partitionable signature scheme in which $\mathtt{Sign} = (\mathsf{S}_1, \mathsf{S}_2)$, and let $D = (\mathtt{DEnc}, \mathtt{DDec})$ be a DEM. Furthermore, let TK and S be SC-composable with shared randomness space $\mathcal{R}$. We assume that the encapsulated-key space of TK and the key space of D is the same (if this is not the case, we can use an appropriate key derivation function).

Then, we construct a signcryption scheme SC as shown in Fig. 7. We note that our scheme allows c_2 in TK and σ'_1 in S to be empty strings. The security of SC is guaranteed by the following theorems. To prove unforgeability, we require *key registration*, as introduced in Section 3.1.

$\mathtt{Setup}_{sc}(1^k)$:	$\mathtt{KG}_S(par)$:
$par_{tk} \leftarrow \mathtt{Setup}_{tk}(1^k)$	Output $(pk_S, sk_S) \leftarrow \mathtt{KG}_{sig}(par_{sig})$.
$par_{sig} \leftarrow \mathtt{Setup}'_{sig}(par_{tk})$	$\mathtt{KG}_R(par)$:
Output $par \leftarrow (par_{tk}, par_{sig})$	Output $(pk_R, sk_R) \leftarrow \mathtt{KG}_{tk}(par_{tk})$.
$\mathtt{SC}(par, sk_S, pk_R, m)$:	$\mathtt{USC}(par, pk_S, sk_R, c)$:
$(r, s) \leftarrow \mathcal{R} \times \mathcal{R}^+_{sig}$, $\mathsf{tag} \leftarrow pk_S$	Parse c as (c_1, c_2, c_3). $\mathsf{tag} \leftarrow pk_S$
$(c_1, \sigma'_1) \leftarrow \mathsf{S}_1(par_{sig}, (r, s))$	$K \leftarrow \mathtt{TDecap}(par_{tk}, sk_R, \mathsf{tag}, (c_1, c_2))$
$\sigma_2 \leftarrow \mathsf{S}_2(par_{sig}, sk_S, (pk_R \| m), (r, s))$	$(m \| \sigma'_1 \| \sigma_2) \leftarrow \mathtt{DDec}(K, c_3)$
$(c_2, K) \leftarrow \mathsf{TE}_2(par_{tk}, pk_R, \mathsf{tag}, r)$	$\sigma \leftarrow (c_1, \sigma'_1, \sigma_2)$
$c_3 \leftarrow \mathtt{DEnc}(K, (m \| \sigma'_1 \| \sigma_2))$	If $\mathtt{SVer}(par_{sig}, pk_S, (pk_R \| m), \sigma) = \top$
Output $c \leftarrow (c_1, c_2, c_3)$.	then output m, otherwise output $\perp$.

Fig. 7. Proposed composition SC from SC-composable TBKEM and signature schemes

Theorem 14. *Assume TK is* IND-tag-CCA *(resp.* IND-stag-CCA*) secure, D is* IND-CCA *secure, and TK and S are SC-composable. Then* SC *is* dM-IND-iCCA *(resp.* fM-IND-iCCA*) secure.*

Theorem 15. *Assume S is* sUF-CMA *(resp.* wUF-CMA*) secure and TK and S are SC-composable. Then* SC *is* dM-sUF-iCMA(KR) *(resp.* dM-wUF-iCMA(KR)*) secure.*

The proofs of the above theorems are given in the full version [35]. Note that, unlike the simple compositions in the previous section, SC achieves strong unforgeability without imposing any restrictions which forces instantiations to be in the random oracle model. Note also that, like StTE, in the unsigncryption process the receiver obtains $\sigma = (c_1, \sigma'_1, \sigma_2)$ which is a publicly verifiable signature of the sender on the sent message m, and hence, the scheme can provide non-repudiation.

7 How to Obtain Tag-Based Primitives

The constructions in Sections 4, 5 and 6 depend on the existence of efficient (T)NIKE schemes, TBE schemes and TBKEM schemes. In this section we will show how existing schemes can be extended to tag-based schemes by exploiting their internal structure. Although this approach is not generic, it is simple, applicable to many of the existing schemes and, importantly, achieves tag-based schemes at practically no additional cost.

7.1 (T)NIKE Schemes in the Random Oracle Model

Consider the *Hashed Diffie-Hellman* (HDH) scheme which is defined as follows: The setup algorithm Setup picks a group $\mathbb{G}$ with prime order p, a generator $g \in \mathbb{G}$, and a hash function $H : \{0,1\}^* \to \{0,1\}^k$. The key generation algorithm KG picks $x \leftarrow \mathbb{Z}_p$ and sets $(pk, sk) \leftarrow (g^x, x)$. Suppose one party's key pair is $(pk_1, sk_1) = (g^x, x)$ and the other's is $(pk_2, sk_2) = (g^y, y)$, and suppose g^x is lexicographically smaller than g^y. Then the shared key algorithm Share outputs $K \leftarrow H(g^x, g^y, g^{xy})$.

It is relatively easy to show that this scheme is secure against active attacks in the random oracle model assuming the gap Diffie-Hellman (GDH) assumption holds in $\mathbb{G}$, using a proof similar to [15]. Furthermore, if the shared key is computed as $K \leftarrow H(g^x, g^y, g^{xy}, \mathsf{tag})$ where tag is a tag given as input to `Share`, the resulting scheme will be a TNIKE scheme which we will denote tHDH. In a similar manner to the HDH scheme, the security of tHDH can be shown assuming the GDH assumption holds in $\mathbb{G}$, using a proof similar to [15].

Lastly, note that a TNIKE scheme secure under the computational Diffie-Hellman (CDH) assumption can be obtained by making a similar modification to the Twin Diffie-Hellman protocol by [15], but at the cost of an increase in computational cost compared to tHDH.

7.2 TBE and TBKEM Schemes

It is possible to generically transform any `IND-CCA` secure PKE scheme into an `IND-tag-CCA` secure TBE scheme, simply by encrypting the tag together with the message [25]. Since a TBE is trivially a TBKEM, this approach also leads to a generic construction of TBKEMs. However, a drawback of this approach is that it leads to ciphertext expansion and possibly inefficient TBKEMs, and since our main concern is efficiency, we take a different approach in the following.

TBE Schemes in the Random Oracle Model. To construct an efficient TBE scheme, we consider the `IND-CCA` secure PKE schemes in the random oracle model which have hybrid structure i.e. they can be rewritten in the KEM/DEM style, and a random oracle is used as a key derivation function for a key of the DEM part. Typical examples of such schemes are the *DHIES* scheme [1] and the *Twin ElGamal* scheme [15]. We can turn such PKE schemes into `IND-tag-CCA` secure TBE schemes simply by inputting a tag into the key derivation function.

Here, as an example, we show in Fig. 8 a tag-based variant of the DHIES scheme which we denote tDHIES. We note that similar modification to the twin ElGamal scheme [15] will result in a corresponding secure tag-based variant (which we denote tTwin).

Since the standard KEM/DEM composition theorem [16] trivially applies to the composition of an `IND-tag-CCA` secure TBKEM and an `IND-CCA` secure DEM, it is sufficient to see that the TBKEM part of the tDHIES is actually `IND-tag-CCA` secure. It is known that the KEM part of the original DHIES is `IND-CCA` secure in the random oracle model based on the GDH assumption [17,21]. Since the proof of the `IND-tag-CCA` security of the TBKEM part of tDHIES is essentially the same as the `IND-CCA` security proof for the ECIES-KEM in [17], we omit the proof here.

TBKEM Schemes in the Standard Model. Here, we consider existing `IND-CCA` secure KEM schemes in the standard model that use a collision resistant hash function (CRHF) or a target CRHF (TCRHF) in the construction of an encapsulation. Specifically, we consider the very efficient and recently proposed schemes [13,26,23,22] which all use a (T)CRHF as a building block (to make

$\mathtt{Setup}_{te}(1^k)$: Pick a group $\mathbb{G}$ (order p) and $g \leftarrow \mathbb{G}$. Pick a DEM $D = (\mathtt{DEnc}, \mathtt{DDEM})$ with key space $\{0,1\}^k$ Pick $H : \{0,1\}^* \to \{0,1\}^k$. Output $par \leftarrow (p, g, \mathbb{G}, D, H)$.	$\mathtt{TEnc}(par, pk, \mathsf{tag}, m)$: $r \leftarrow \mathbb{Z}_p, \quad c_1 \leftarrow g^r, \quad K \leftarrow H(\mathsf{tag}\|\|c_1\|\|X^r)$ $c_2 \leftarrow \mathtt{DEnc}(K, m)$ $c \leftarrow (c_1, c_2)$ Output (c, K).
$\mathtt{KG}_{te}(par)$: $x \leftarrow \mathbb{Z}_p$, $X \leftarrow g^x$ Output $(pk, sk) \leftarrow (X, x)$.	$\mathtt{TDec}(par, sk, \mathsf{tag}, c)$: Parse c as (c_1, c_2). $\quad K \leftarrow H(\mathsf{tag}\|\|c_1\|\|c_1^x)$ Output $m \leftarrow \mathtt{DDec}(K, c_2)$.

Fig. 8. A TBE scheme based on the DHIES scheme (tDHIES)

$\mathtt{Setup}_{tk}(1^k)$: Pick bilinear groups $(\mathbb{G}, \hat{\mathbb{G}}, \mathbb{G}_T)$ (order p) with $e : \mathbb{G} \times \hat{\mathbb{G}} \to \mathbb{G}_T$ and $\psi : \hat{\mathbb{G}} \to \mathbb{G}$ Pick $\hat{g} \leftarrow \hat{\mathbb{G}}$, and set $g \leftarrow \psi(\hat{g})$. Pick a CRHF $\mathsf{CR} : \{0,1\}^* \to \{0,1\}^n$. Output $par \leftarrow (p, \mathbb{G}, \mathbb{G}_T, e, \psi, g, \hat{g}, \mathsf{CR})$.	$\mathtt{TEncap}(par, pk, \mathsf{tag})$: $r \leftarrow \mathbb{Z}_p, \quad c_1 \leftarrow g^r, \quad t \leftarrow \mathsf{CR}(\mathsf{tag}\|\|c_1)$ Let t be an n-bit string $t_1\|\|t_2\|\| \ldots \|\|t_n$ $c_2 \leftarrow (U' \prod_{i=1}^n U_i^{t_i})^r$ $c \leftarrow (c_1, c_2), \quad K \leftarrow \mathsf{Z}^r$ Output (c, K).
$\mathtt{KG}_{tk}(par)$: $u', u_1, \ldots, u_n \leftarrow \mathbb{Z}_p$ $U' \leftarrow g^{u'}$, $U_i \leftarrow g^{u_i}$ for $1 \le i \le n$ $\alpha \leftarrow \mathbb{Z}_p, \quad \hat{h} \leftarrow \hat{g}^\alpha, \quad \mathsf{Z} \leftarrow e(g, \hat{g})^\alpha$ $pk \leftarrow (\mathsf{Z}, U', U_1, \ldots, U_n)$ $sk \leftarrow (\hat{h}, u', u_1, \ldots, u_n)$ Output (pk, sk).	$\mathtt{TDecap}(par, sk, \mathsf{tag}, c)$: Parse c as (c_1, c_2). $t \leftarrow \mathsf{CR}(\mathsf{tag}\|\|c_1)$ Let t be an n-bit string $t_1\|\|t_2\|\| \ldots \|\|t_n$ If $c_2 = c_1^{u' + \sum_{i=1}^n u_i t_i}$ Output $K \leftarrow e(c_1, \hat{h})$ Otherwise output $\perp$.

Fig. 9. TBKEM scheme based on the Boyen-Mei-Waters PKE (tBMW1)

[23] IND-CCA secure, we have to apply the technique from [4]). In these schemes, if we simply add a tag as an additional input to the hash function, we can achieve secure TBKEM schemes. As an example, we show in Fig. 9 a partitionable TBKEM scheme obtained from the practical PKE proposed by Boyen, Mei, and Waters [13] which we denote tBMW1 (note that the original scheme is a PKE scheme but here, we turn it into a TBKEM scheme).

Since the security proof is essentially the proof of the original BMW PKE scheme (whose proof is almost identical to that of the Waters IBE [45] which is adaptive identity chosen plaintext secure), the details are omitted here.

Several other KEMs share a similar structure to the Boyen *et al.* KEM [12] (e.g. [26,23,22,21]), and can be modified in a similar way to achieve TBKEMs. However, whether IND-tag-CCA or IND-stag-CCA security is achieved is dependent on how the original KEM is proved secure. In particular, the TBKEMs obtained from [26,23,22,21] will only achieve IND-stag-CCA security.

8 Concrete SC-Composable Schemes

We will now introduce a number of signature/TBKEM pairs which are SC-composable, using the TBKEMs introduced in the previous section. Consider the

TBKEM tBMW1 shown in Fig. 9. The scheme is partitionable with the algorithms $\mathsf{TE}_1(par, r) = g^r$ and $\mathsf{TE}_2(par, pk_R, \mathsf{tag}, r) = (c_2, K)$ where $c_2 \leftarrow (U' \prod_{i=1}^{n} U_i^{t_i})^r$, $K \leftarrow \mathsf{Z}^r$ and $t \leftarrow \mathsf{CR}(\mathsf{tag}||g^r)$. An example of a suitable signature scheme to combine with this TBKEM is the scheme by Waters [45] (Waters). Here, we assume that Waters is implemented with the same bilinear groups as tBMW1 in Fig. 9. Signatures are of the form $\sigma = (g^r, g^\alpha \cdot \psi(\hat{V}' \prod_{i=1}^{n} \hat{V}_i^{m_i})^r) \in (\mathbb{G})^2$ where $g^\alpha \in \mathbb{G}$ and $(\hat{V}', \hat{V}_1, \ldots, \hat{V}_n) \in (\hat{\mathbb{G}})^{n+1}$ are elements of the private and public signer key, sk_S and pk_S, respectively, and m_i is the i-th bit of the message m (see [45] for a full description of the scheme). Furthermore, the scheme is partitionable with $\mathsf{S}_1(par, r) = g^r$ and $\mathsf{S}_2(par, sk_S, m, r) = g^\alpha \cdot \psi(\hat{V}' \prod_{i=1}^{n} \hat{V}_i^{m_i})^r$, where $r \in \mathbb{Z}_p$.

It is relatively easy to check that the two schemes satisfy the requirements about compatible setup (property 1) and shared randomness (property 2) of Definition 13 with shared randomness space $\mathbb{Z}_p$. Furthermore the algorithms TE_2' and S_2' for the scheme are defined as $\mathsf{TE}_2'(sk, \mathsf{tag}, g^r) = ((g^r)^{u' + \sum_{i=1}^{n} u_i t_i}, e(g^r, \hat{g}^\alpha))$ and $\mathsf{S}_2'(sk, m, g^r) = g^\alpha \cdot (g^r)^{u' + \sum_{i=1}^{n} u_i m_i}$, and satisfy the requirement about ciphertext/signature simulatability (property 3). Taking into account that tBMW1 is IND-tag-CCA secure and that Waters is wUF-CMA secure, Theorem 14 and 15 yields that the signcryption scheme SC shown in Fig. 7 is dM-IND-iCCA and dM-wUF-iCMA(KR) secure when instantiated with these schemes.

However, there are many other SC-composable pairs. For example, if a strongly unforgeable signcryption scheme is desired, Waters signatures can be replaced by the sUF-CMA secure variant proposed by Boneh *et al.* [11] (BSW). Alternatively, the signatures by Camenisch *et al.* [14] (CL) can be used to achieve a scheme with compact public sender keys (this scheme can furthermore be made sUF-CMA secure using the techniques from [11]). Likewise, the TBKEM can be replaced by any of the TBKEMs mentioned in the previous section to achieve signcryption schemes with various properties. Note that any combination of the mentioned signature schemes and TBKEMs will be SC-composable (see the full version [35] for details).

9 Comparison

In Fig. 10, we list the achieved security notions, underlying security assumptions and computational and ciphertext overhead for previously proposed signcryption schemes as well as the constructions discussed in this paper. All schemes are instantiated to obtain minimal ciphertext and computational overhead. Specifically, we assume that an IND-CCA secure DEM has no ciphertext overhead (i.e. is length preserving) and IND-CPA and IND-CCA secure SKE have ciphertext overheads which are of the size $|IV|$ and $|IV| + |\mathtt{MAC}|$ [7], respectively. In the original schemes of LYWDC [28] and LQ [29], the public sender key is included as part of the plaintext. However, this is only needed when considering anonymity, and we leave out the sender key from the plaintext in these schemes. Dent [18] and GBN [20] require a "signcryption DEM" which is a DEM that satisfies both IND-CCA and INT-CTXT security. To achieve this we assume that the Encrypt-then-MAC approach is used as discussed in [18].

Scheme	RO	Confidentiality	Unforgeability	Comp. Cost sc	Comp. Cost usc	Overhead Elements	Overhead Bits
Dent [18]	Yes	IND-oCCA / CDH	sUF-oCMA / CDH	[2,0;0]	[1,0;0]	$\|\mathbb{G}_e\| + \|\mathtt{MAC}\|$	240
BD [8]	Yes	IND-oCCA / GDH	sUF-iCMA / CDH	[3,0;0]	[0,2;0]	$\|\mathbb{G}_e\| + 2\|\mathbb{Z}_p\|$	480
GBN [20] + HMQV [27]	Yes	sM-IND-iCCA / CDH	sM-sUF-oCMA / CDH	[2,0;0]	[0,1;0]	$\|\mathbb{G}_e\| + \|\mathtt{MAC}\|$	240
Zheng [47,5,6]	Yes	fM-IND-oCCA / GDH	dM-sUF-iCMA / GDL	[1,0;0]	[1,1;0]	$2\|\mathbb{Z}_p\|$	320
LQ [29]	Yes	dM-IND-iCCA / co-CDH	dM-sUF-iCMA / co-CDH	[3,0;0]	[1,0;2]	$2\|\mathbb{G}_p\|$	342
LYWDC [28]	Yes	dM-IND-iCCA / co-CDH	dM-sUF-iCMA / co-CDH	[3,0;0]	[1,0;2]	$2\|\mathbb{G}_p\|$	342
Tan1 [42] + BB [9]	No	dM-IND-iCCA / DDH	sUF-iCMA / q-SDH	[3,1;0]	[0,2;1]	$2\|\mathbb{G}_e\| + \|\mathbb{G}_p\| + \|\mathbb{Z}_p\| + \|\mathtt{MAC}\|$	731
Tan2 [43] + BB [9]	No	dM-IND-iCCA / DDH	sUF-iCMA / q-SDH	[4,1;0]	[1,2;1]	$3\|\mathbb{G}_e\| + \|\mathbb{G}_p\| + \|\mathbb{Z}_p\|$	811
Tan3 [44]	No	dM-IND-iCCA / DBDH	dM-sUF-iCMA(KR) / q-SDH	[3,2;0]	[2,1;4]	$3\|\mathbb{G}_p\| + 2\|\mathbb{Z}_p\|$	833
K&SE(HDH)	Yes	fM-IND-oCCA / GDH	fM-sUF-oCMA / GDH	[1,0;0]	[1,0;0]	$\|IV\| + \|\mathtt{MAC}\|$	160
TK&SEtS(tHDH,BLS)	Yes	fM-IND-oCCA / GDH, co-CDH	dM-sUF-iCMA / co-CDH	[2,0;0]	[1,0;2]	$\|IV\| + \|\mathbb{G}_p\|$	251
TEtK&M(HDH,tDHIES)	Yes	dM-IND-iCCA / GDH	fM-sUF-oCMA / GDH	[3,0;0]	[2,0;0]	$\|\mathbb{G}_e\| + \|\mathtt{MAC}\|$	240
TEtS(tDHIES,BLS)	Yes	dM-IND-iCCA / GDH	dM-sUF-iCMA / co-CDH	[3,0;0]	[1,0;2]	$\|\mathbb{G}_e\| + \|\mathbb{G}_p\|$	331
TEtS(tTwin,BLS)	Yes	dM-IND-iCCA / CDH	dM-sUF-iCMA / co-CDH	[4,0;0]	[2,0;2]	$\|\mathbb{G}_e\| + \|\mathbb{G}_p\|$	331
StTE(tBMW1,Waters)	No	dM-IND-iCCA / DBDH	dM-wUF-iCMA / co-CDH	[5,0;0]	[1,0;3]	$4\|\mathbb{G}_p\|$	684
StTE(tBMW1,BB)	No	dM-IND-iCCA / DBDH	dM-wUF-iCMA / q-SDH	[4,0;0]	[1,1;2]	$3\|\mathbb{G}_p\| + \|\mathbb{Z}_p\|$	673
SC(tBMW2,CL)	No	fM-IND-iCCA / DBDH	dM-wUF-iCMA(KR) / LRSW	[4,1;0]	[1,0;6]	$4\|\mathbb{G}_p\|$	684
SC(tBMW2,CL')	No	fM-IND-iCCA / DBDH	dM-sUF-iCMA(KR) / LRSW	[4,2;0]	[1,1;6]	$4\|\mathbb{G}_p\| + \|\mathbb{Z}_p\|$	844
SC(tHaKu,Waters)	No	fM-IND-iCCA / CDH	dM-wUF-iCMA(KR) / co-CDH	[4+k,0;0]	[2+k,0;2]	$4\|\mathbb{G}_p\|$	684
SC(tHaKu,BSW)	No	fM-IND-iCCA / CDH	dM-sUF-iCMA(KR) / co-CDH	[4+k,1;0]	[2+k,1;2]	$4\|\mathbb{G}_p\| + \|\mathbb{Z}_p\|$	844
SC(tBMW1,Waters)	No	dM-IND-iCCA / DBDH	dM-wUF-iCMA(KR) / co-CDH	[4,0;0]	[1,0;3]	$3\|\mathbb{G}_p\|$	513
SC(tBMW1,BSW)	No	dM-IND-iCCA / DBDH	dM-sUF-iCMA(KR) / co-CDH	[4,1;0]	[1,1;3]	$3\|\mathbb{G}_p\| + \|\mathbb{Z}_p\|$	673

Fig. 10. Comparison of existing and proposed signcryption schemes. Column "Comp. Cost" lists the computational overhead for signcryption (sc) and unsigncryption (usc), where $[a, b; c]$ denotes a exponentiations, b multi-exponentiations and c pairing computations (multiplications, computation costs of hash functions and symmetric key primitives are ignored). In the overhead column, $|\mathbb{G}_e|$, $|\mathbb{G}_p|$, $|\mathbb{Z}_p|$, $|IV|$, and $|\mathtt{MAC}|$ denote the size of a group element on an ordinary elliptic curve, that of an elliptic curve equipped with an asymmetric pairing, the size of an exponent, the size of an initialization vector for SKE, and the size of a MAC tag, respectively. When instantiated to achieve 80 bits of security, we set $|\mathbb{G}_e| = |\mathbb{Z}_p| = |H| = 160$ bits, $|\mathbb{G}_p| = 171$ bits, $|\mathbb{G}_{sp}| = 237$ bits, and $|\mathtt{MAC}| = |IV| = 80$ bits. tBMW2 and tHaKu denote the TBKEMs obtained from [12] and [22] as explained in Section 7.2.

The scheme K&SE(HDH) has the lowest ciphertext and computational overhead of all signcryption schemes, while providing outsider multi-user security. This improves upon the Dent scheme [18] which is furthermore only shown to be secure in the two-user setting. If unforgeability against insiders is required, the scheme TK&SEtS(tHDH,BLS) provides the lowest ciphertext overhead, but the Zheng scheme [47,5,6] has lower computational cost. On the other hand, if confidentiality against insiders is required (but only outsider unforgeability), the schemes TEtK&M (HDH,tDHIES) and GBN [20] provides the same ciphertext overhead but GBN provides slightly lower computational overhead. However, GBN is only shown to be secure in the weaker static multi-user model which implies key registration, and in this aspect we consider TEtK&M(HDH,tDHIES) as an improvement upon GBN.

Considering schemes which provides full insider security, TEtS(tDHIES,BLS) improves upon LYWDC [28] and LQ [29] by providing slightly lower ciphertext overhead while having practically the same computational cost (an IND-CCA secure DEM vs. a one-time pad). The ciphertext overhead is in fact lower than BD [8] and only 11 bits larger than Zheng although these schemes provides a lower level of security.

The schemes based on SC-composable TBKEMs and signatures improves upon the previous standard model schemes by providing both lower ciphertext and computational overhead. The lowest overhead is achieved by SC(tBMW1,Waters) (and SC(tBMW2,Waters)). However, if strong unforgeability is desired, the slightly less efficient SC(tBMW1,BSW) is required. The only drawback of the schemes based on SC-composability is that key registration is required to guarantee unforgeability (note that previous standard model schemes requires key registration as well). If key registration is not feasible, the most efficient scheme would be StTE(tBMW1,Waters) or StTE(tBMW1,BB) where BB denotes the short signature scheme by Boneh *et al.* [9]. Lastly note that the schemes based on tBMW2 TBKEM and the signature scheme by Camenisch *et al.* [14] (CL) (or CL [14] modified with the technique from [11], denoted by CL') have the advantage of compact public sender and receiver keys.

References

1. Abdalla, M., Bellare, M., Rogaway, P.: The oracle Diffie-Hellman assumptions and an analysis of DHIES. In: Naccache, D. (ed.) CT-RSA 2001. LNCS, vol. 2020, pp. 143–158. Springer, Heidelberg (2001)
2. Abe, M., Gennaro, R., Kurosawa, K., Shoup, V.: Tag-KEM/DEM: A new framework for hybrid encryption and a new analysis of Kurosawa-Desmedt KEM. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 128–146. Springer, Heidelberg (2005)
3. An, J.H., Dodis, Y., Rabin, T.: On the security of joint signature and encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 83–107. Springer, Heidelberg (2002)
4. Baek, J., Galindo, D., Susilo, W., Zhou, J.: Constructing strong KEM from weak KEM (or how to revive the KEM/DEM framework). In: Ostrovsky, R., De Prisco, R., Visconti, I. (eds.) SCN 2008. LNCS, vol. 5229, pp. 358–374. Springer, Heidelberg (2008)

5. Baek, J., Steinfeld, R., Zheng, Y.: Formal proofs for the security of signcryption. In: Naccache, D., Paillier, P. (eds.) PKC 2002. LNCS, vol. 2274, pp. 80–98. Springer, Heidelberg (2002)
6. Baek, J., Steinfeld, R., Zheng, Y.: Formal proofs for the security of signcryption. J. Cryptology 20(2), 203–235 (2007)
7. Bellare, M., Namprempre, C.: Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In: Okamoto, T. (ed.) ASIACRYPT 2000. LNCS, vol. 1976, pp. 531–545. Springer, Heidelberg (2000)
8. Bjørstad, T.E., Dent, A.W.: Building better signcryption schemes with tag-KEMs. In: Yung, M., Dodis, Y., Kiayias, A., Malkin, T.G. (eds.) PKC 2006. LNCS, vol. 3958, pp. 491–507. Springer, Heidelberg (2006)
9. 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)
10. Boneh, D., Lynn, B., Shacham, H.: Short signatures from the Weil pairing. J. Cryptology 17(4), 297–319 (2004)
11. Boneh, D., Shen, E., Waters, B.: Strongly unforgeable signatures based on computational Diffie-Hellman. In: Yung, M., Dodis, Y., Kiayias, A., Malkin, T.G. (eds.) PKC 2006. LNCS, vol. 3958, pp. 229–240. Springer, Heidelberg (2006)
12. Boyen, X., Mei, Q., Waters, B.: Direct chosen ciphertext security from identity-based techniques. In: Cryptology ePrint Archive, Report 2005/288 (2005)
13. Boyen, X., Mei, Q., Waters, B.: Direct chosen ciphertext security from identity-based techniques. In: ACM CCS 2005, pp. 320–329. ACM, New York (2005)
14. Camenisch, J., Lysyanskaya, A.: Signature schemes and anonymous credentials from bilinear maps. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 56–72. Springer, Heidelberg (2004)
15. Cash, D., Kiltz, E., Shoup, V.: The twin Diffie-Hellman problem and applications. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 127–145. Springer, Heidelberg (2008)
16. Cramer, R., Shoup, V.: Design and analysis of practical public-key encryption schemes secure against chosen ciphertext attack. SIAM J. Computing 33(1), 167–226 (2003)
17. Dent, A.W.: ECIES-KEM vs. PSEC-KEM, Technical Report NES/DOC/RHU/WP5/028/2 (2002)
18. Dent, A.W.: Hybrid signcryption schemes with outsider security. In: Zhou, J., López, J., Deng, R.H., Bao, F. (eds.) ISC 2005. LNCS, vol. 3650, pp. 203–217. Springer, Heidelberg (2005)
19. Diffie, W., Hellman, M.E.: New directions in cryptography. IEEE Transactions on Information Theory IT-22(6), 644–654 (1976)
20. Gorantla, M.C., Boyd, C., Nieto, J.M.G.: On the connection between signcryption and one-pass key establishment. In: Galbraith, S.D. (ed.) Cryptography and Coding 2007. LNCS, vol. 4887, pp. 277–301. Springer, Heidelberg (2007)
21. Hanaoka, G., Imai, H., Ogawa, K., Watanabe, H.: Chosen ciphertext secure public key encryption with a simple structure. In: Matsuura, K., Fujisaki, E. (eds.) IWSEC 2008. LNCS, vol. 5312, pp. 20–33. Springer, Heidelberg (2008)
22. Hanaoka, G., Kurosawa, K.: Efficient chosen ciphertext secure public key encryption under the computational Diffie-Hellman assumption. In: Pieprzyk, J. (ed.) ASIACRYPT 2008. LNCS, vol. 5350, pp. 308–325. Springer, Heidelberg (2008)

23. Hofheinz, D., Kiltz, E.: Secure hybrid encryption from weakened key encapsulation. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 553–571. Springer, Heidelberg (2007)
24. Jeong, I.R., Jeong, H.Y., Rhee, H.S., Lee, D.H., Lim, J.I.: Provably secure encrypt-then-sign composition in hybrid signcryption. In: Lee, P.J., Lim, C.H. (eds.) ICISC 2002. LNCS, vol. 2587, pp. 16–34. Springer, Heidelberg (2003)
25. Kiltz, E.: Chosen-ciphertext security from tag-based encryption. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 581–600. Springer, Heidelberg (2006)
26. Kiltz, E.: Chosen-ciphertext secure key-encapsulation based on gap hashed Diffie-Hellman. In: Okamoto, T., Wang, X. (eds.) PKC 2007. LNCS, vol. 4450, pp. 282–297. Springer, Heidelberg (2007)
27. Krawczyk, H.: HMQV: A high-performance secure Diffie-Hellman protocol. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 546–566. Springer, Heidelberg (2005)
28. Li, C.K., Yang, G., Wong, D.S., Deng, X., Chow, S.S.M.: An efficient signcryption scheme with key privacy. In: López, J., Samarati, P., Ferrer, J.L. (eds.) EuroPKI 2007. LNCS, vol. 4582, pp. 78–93. Springer, Heidelberg (2007)
29. Libert, B., Quisquater, J.-J.: Efficient signcryption with key privacy from gap Diffie-Hellman groups. Updated version of[31], `http://www.dice.ucl.ac.be/~libert/`
30. Libert, B., Quisquater, J.-J.: Improved signcryption from q-Diffie-Hellman problems. Updated version of[32], `http://www.dice.ucl.ac.be/~libert/`
31. Libert, B., Quisquater, J.-J.: Efficient signcryption with key privacy from gap Diffie-Hellman groups. In: Bao, F., Deng, R., Zhou, J. (eds.) PKC 2004. LNCS, vol. 2947, pp. 187–200. Springer, Heidelberg (2004)
32. Libert, B., Quisquater, J.-J.: Improved signcryption from q-Diffie-Hellman problems. In: Blundo, C., Cimato, S. (eds.) SCN 2004. LNCS, vol. 3352, pp. 220–234. Springer, Heidelberg (2005)
33. Ma, C.: Efficient short signcryption scheme with public verifiability. In: Lipmaa, H., Yung, M., Lin, D. (eds.) INSCRYPT 2006. LNCS, vol. 4318, pp. 118–129. Springer, Heidelberg (2006)
34. MacKenzie, P.D., Reiter, M.K., Yang, K.: Alternatives to non-malleability: Definitions, constructions, and applications (extended abstract). In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 171–190. Springer, Heidelberg (2004)
35. Matsuda, T., Matsuura, K., Schuldt, J.C.N.: Efficient constructions of signcryption schemes and signcryption composability, `http://eprint.iacr.org`
36. Shoup, V.: Sequences of games: a tool for taming complexity in security proofs. Cryptology ePrint Archive, Report 2004/332 (2004)
37. Shoup, V., Gennaro, R.: Securing threshold cryptosystems against chosen ciphertext attack. In: Nyberg, K. (ed.) EUROCRYPT 1998. LNCS, vol. 1403, pp. 1–16. Springer, Heidelberg (1998)
38. Tan, C.H.: On the security of signcryption scheme with key privacy. IEICE Transactions 88-A(4), 1093–1095 (2005)
39. Tan, C.H.: Analysis of improved signcryption scheme with key privacy. Inf. Process. Lett. 99(4), 135–138 (2006)
40. Tan, C.H.: Security analysis of signcryption scheme from q-Diffie-Hellman problems. IEICE Transactions 89-A(1), 206–208 (2006)
41. Tan, C.H.: Forgery of provable secure short signcryption scheme. IEICE Transactions 90-A(9), 1879–1880 (2007)

42. Tan, C.H.: Insider-secure hybrid signcryption scheme without random oracles. In: ARES 2007, pp. 1148–1154. IEEE Computer Society, Los Alamitos (2007)
43. Tan, C.H.: Insider-secure signcryption KEM/tag-KEM schemes without random oracles. In: ARES 2008, pp. 1275–1281. IEEE Computer Society, Los Alamitos (2008)
44. Tan, C.H.: Signcryption scheme in multi-user setting without random oracles. In: Matsuura, K., Fujisaki, E. (eds.) IWSEC 2008. LNCS, vol. 5312, pp. 64–82. Springer, Heidelberg (2008)
45. Waters, B.: Efficient identity-based encryption without random oracles. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 114–127. Springer, Heidelberg (2005)
46. Yang, G., Wong, D.S., Deng, X.: Analysis and improvement of a signcryption scheme with key privacy. In: Zhou, J., López, J., Deng, R.H., Bao, F. (eds.) ISC 2005. LNCS, vol. 3650, pp. 218–232. Springer, Heidelberg (2005)
47. Zheng, Y.: Digital signcryption or how to achieve cost(signature & encryption) << cost(signature) + cost(encryption). In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 165–179. Springer, Heidelberg (1997)