Certificateless Encryption Schemes Strongly Secure in the Standard Model

Alexander W. Dent[1], Benoît Libert[2], and Kenneth G. Paterson[1]

[1] Information Security Group,
Royal Holloway, University of London (United Kingdom)
[2] UCL, Microelectronics Laboratory, Crypto Group (Belgium)

Abstract. This paper presents the first constructions for certificateless encryption (CLE) schemes that are provably secure against strong adversaries in the standard model. It includes both a generic construction for a strongly secure CLE scheme from any passively secure scheme as well as a concrete construction based on the Waters identity-based encryption scheme.

Keywords: certificateless encryption, standard model, strong security.

1 Introduction

Certificateless public key cryptography (CL-PKC), as proposed by Al-Riyami and Paterson [1], represents an interesting and potentially useful balance between identity-based cryptography and public key cryptography based on certificates. It eliminates the key escrow associated with identity-based cryptography without requiring the introduction of certificates, which pose many operational difficulties in PKIs. The main idea of CL-PKC is that a user Alice combines two key components to form her private key: one component (the partial private key, PPK) is generated by a Key Generation Centre (KGC) using a master secret, and another component (the secret value) is generated by the user herself. The user also publishes a public key derived from her secret value; a party who wishes to encrypt to Alice only needs to have Alice's identity and public key along with the KGC's public parameters. One novel aspect of CL-PKC is the modelling of adversaries who are capable of replacing the public keys of users with keys of their choice. This is necessary because there are no certificates to authenticate users' public keys in CL-PKC.

The topic of certificateless cryptography has undergone quite rapid development, with many schemes being proposed for encryption (CLE) [1,3,6,12,25] and signatures (CLS) [1,20,22,32,35]. One notable feature has been the development of a number of alternative security models for CLE that are substantially weaker than the original model of [1]. These different models are summarised by Dent [13]. In the model of [1], the attacker is of one of two types. The Type I attacker models an "outsider" adversary, who can replace the public keys of users, obtain PPKs and private keys, and make decryption queries. The Type II attacker models an "honest-but-curious" KGC who is given the master secret (and can therefore

R. Cramer (Ed.): PKC 2008, LNCS 4939, pp. 344–359, 2008.
© International Association for Cryptologic Research 2008

generate any PPK), can obtain private keys and make decryption queries, but is trusted not to replace any public keys. (We actually use a slightly stronger model of security for Type II attackers, in which the attacker can replace public keys providing that they do not allow the attacker to trivially break the scheme.)

In their original security model, Al-Riyami and Paterson chose to make the Type I adversary as strong as possible, insisting in their model that a challenger should correctly respond to decryption queries even if the public key of a user had been replaced. This is called a Strong Type I attacker in [13]. Currently, the only published CLE schemes that have been proven secure against strong Type I adversaries [1,25] make use of the random oracle model [4]. Notably, Libert and Quisquater [25] provide a generic construction which converts a CLE scheme secure against passive adversaries (who do not have access to a decryption oracle) into a scheme secure against strong adversaries, using a Fujisaki-Okamoto-style conversion [17]. This conversion allows decryption queries to be handled using a form of knowledge extraction, but does require the use of random oracles.

Related Work

In 2003, Gentry [19] introduced a different but related concept named certificate based encryption (CBE). This approach is closer to the context of a traditional PKI model as it involves a certification authority (CA) providing an efficient implicit certification service for clients' public keys.

Subsequent works [33,31] considered the relations between identity-based (IBE), certificate based (CBE) and certificateless encryption schemes (CLE) and established a result of essential equivalence [33] between the three primitives. The generic transformations of [33,31] do not use random oracles but those results do not hold in the full security model developed in [1] for CLE schemes; indeed, they were even shown not to hold in relaxed CLE models [18].

In [15], Dodis and Katz described generic methods to construct IND-CCA secure multiple-encryption schemes from public key encryption schemes which are individually IND-CCA. They proved that their methods apply to the design of certificate-based encryption schemes [19] and yield CBE schemes without random oracles. Because of the strong properties required of decryption oracles in [1], these techniques do not directly apply in the present context. In security proofs, the technical difficulty is that the simulator does not know the secret value of entities whose public key was replaced. In other words, the constructions of [15] are not designed to handle decryption queries for arbitrary public keys chosen "on-the-fly" by adversaries who may not even know the matching secret as in the present context.

Other authors [26] have also recently attempted to address the problem of designing certificateless cryptosystems (or related primitives) in the standard model. However their results are not presented in the full model of [1]. In particular, the recent work of Huang and Wong [21] constructs a certificateless encryption scheme that is secure in the standard model but does not permit a Strong Type I adversary.

Finally, a recently initiated research direction considers authorities that maliciously generate system-wide parameters [2]. As we shall see, the model of [2]

makes it even more difficult to devise schemes that are provably secure in the standard model. Neither of the schemes we present are secure against adversaries that maliciously generate the system-wide parameters.

Our Contributions

We make two contributions which resolve questions raised by the above debate concerning CLE security models.

Firstly, we present a generic construction for strongly secure CLE. Our construction uses any CLE scheme and any normal public key encryption (PKE) scheme as components, but these only need to be secure against passive adversaries. In contrast to [25], our construction does not intrinsically require the use of random oracles. Instead, we use an extension of the techniques of Naor-Yung [27] and Sahai [29]; however, some additional ideas are needed to handle decryption queries for adversarially-selected public keys. As it makes use of non-interactive zero-knowledge (NIZK) proofs for general statements in NP, our generic construction cannot be regarded as being practical.

Secondly, we provide the first concrete and efficient construction for a CLE scheme that is secure in the standard model against strong adversaries. In fact, our scheme is secure against both Strong Type I attackers and Strong Type II adversaries. The latter represents a natural strengthening of the original Type II adversary introduced in [1]. The construction is based upon the Waters identity-based encryption (IBE) scheme, modifying this scheme using ideas from [1]. The scheme enjoys relatively short public keys and ciphertexts; its security is based on the hardness of a slight and natural generalisation of the DBDH problem.

Why Consider Strong Decryption Oracles?

There has been some debate on whether the Strong Type I and Strong Type II security models correctly model the security capabilities of an attacker against a certificateless encryption scheme [1,6,12,21]. A full discussion of this issue is given in the survey by Dent [13]. It can be argued that an attacker should be given access to an oracle if it supplies information that an attacker might be able to obtain in real life. For example, a decryption oracle provides information about a message that an attacker might be able to obtain by observing how a system behaves after receiving and decrypting a ciphertext or by bribing/threatening the user who received a ciphertext. In certificateless encryption, it is necessary to model the adversary's ability to fool a sender into using the wrong public key when encrypting a message, because public keys are not supported by certificates. This is done by allowing the adversary to replace public keys at will in the model. But there is no reason to suppose that a recipient would use anything other than its own, original private key when decrypting. So there is no practical reason to require that a decryption oracle for a replaced public key should be available to the attacker.

However, we still believe that the results of this paper are of theoretical interest to the research community, even if they are not practically relevant. There are several reasons for this:

– The strong models have been widely used in the previous papers and the question of whether it is possible to construct a scheme that is secure in the

Strong Type I and Strong Type II models without using the random oracle methodology has been widely discussed. Indeed, it has even been conjectured that it was impossible to construct schemes that are both Strong Type I and Strong Type II secure in the standard model. In this paper, we show this conjecture to be false.
- Even if the strong model is not of practical interest, security in this model does guarantee security in the weaker, but more practically relevant, security models. Hence, at a basic level, this paper can be seen to be proving the security of several certificateless encryption schemes in the standard model (assuming honest-but-curious KGCs). Of particular interest is the generic construction presented in Section 3, which demonstrates that certificateless encryption schemes can be constructed from generic assumptions.
- Lastly, our work demonstrates that it is possible for a polynomial-time scheme to be secure in a model that allows the attacker access to oracles that compute non-polynomial-time functions (in this case computing the decryptions of ciphertexts created using arbitrary public keys). We believe that the idea of considering the security of schemes in non-polynomial-time models to be theoretically interesting.

2 Preliminaries

2.1 Notation

We use the following notation. Let $\emptyset$ denote the empty bitstring. If $\mathcal{A}$ is a deterministic algorithm, then $y \leftarrow \mathcal{A}(x)$ denotes the assignment to y of the output of $\mathcal{A}$ when run on the input x. If $\mathcal{A}$ is a randomised algorithm, then $y \xleftarrow{\$} \mathcal{A}(x)$ the assignment to y of the output of $\mathcal{A}$ when run on the input x with a fresh random tape. We let $y \leftarrow A(x; r)$ denote the assignment to y of the output of $\mathcal{A}$ when run on the input x with the random tape r. If $\mathcal{A}$ is a probabilistic polynomial-time (PPT) algorithm, then we may assume that r is of polynomial length. If S is a finite set, then $y \xleftarrow{\$} S$ denotes the random generation of an element $x \in S$ using the uniform distribution. A function $\nu : \mathbb{N} \rightarrow [0, 1]$ is said to be *negligible* if for all $c \in \mathbb{N}$ there exists a $k_c \in \mathbb{N}$ such that $\nu(k) < k^{-c}$ for all $k > k_c$.

2.2 Certificateless Encryption Schemes

The notion of a certificateless encryption scheme was introduced by Al-Riyami and Paterson [1]. A certificateless public-key encryption scheme is defined by seven probabilistic, polynomial-time algorithms:

- `Setup`: takes as input a security parameter 1^k and returns the master private key msk and the master public key mpk. This algorithm is run by a KGC to initially set up a certificateless system.
- `Extract`: takes as input the master public key mpk, the master private key msk, and an identifier $\mathsf{ID} \in \{0, 1\}^*$. It outputs a partial private key d_{ID}. This

algorithm is run by a KGC once for each user, and the corresponding partial private key is distributed to that user in a suitably secure manner.
- `SetSec`: given the master public key mpk and an entity's identifier ID as input, and outputs a secret value x_{ID} for that identity. This algorithm is run once by the user.
- `SetPriv`: takes as input the master public key mpk, an entity's partial private key d_{ID} and an entity's secret value x_{ID}. It outputs the full private key sk_{ID} for that user. This algorithm is run once by the user.
- `SetPub`: given the master public key mpk and an entity's secret value x_{ID}, this algorithm outputs a public key $pk_{\mathsf{ID}} \in \mathcal{PK}$ for that user. This algorithm is run once by the user and the resulting public key is widely and freely distributed. The public-key space $\mathcal{PK}$ is defined using mpk and is assumed to be publicly recognisable: given mpk, public keys having a matching private key should be easily distinguishable from ill-formed public keys.
- `Encrypt`: this algorithm takes as input the master public key mpk, a user's identity ID, a user's public key $pk_{\mathsf{ID}} \in \mathcal{PK}$ and a message $m \in \mathcal{M}$. It outputs either a ciphertext $C \in \mathcal{C}$ or the error symbol $\perp$.
- `Decrypt`: this algorithm takes as input the master public key mpk, a user's private key sk_{ID} and a ciphertext $C \in \mathcal{C}$. It returns either a message $m \in \mathcal{M}$ or the error symbol $\perp$.

We insist that all certificateless encryption schemes satisfy the obvious correctness conditions (that decryption "undoes" encryption).

Dent [13] has surveyed the numerous different security models proposed for certificateless encryption. In this paper, we will only be concerned with the Strong Type I and Strong Type II security definitions. Both of these security models consider attack games that extend the standard IND-CCA attack game for public-key encryption. In both games, we are concerned with the difference in probability

$$Adv_{\mathcal{A}}^{\texttt{CL-CCA-X}}(k) = |Pr[Expt_{\mathcal{A}}^{\texttt{CL-CCA-X}}(0,k) = 1] - Pr[Expt_{\mathcal{A}}^{\texttt{CL-CCA-X}}(1,k) = 1]|$$

for $\texttt{X} \in \{\texttt{I}, \texttt{II}\}$ where $\mathcal{A}$ is any PPT adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ and the experiment $Expt_{\mathcal{A}}^{\texttt{CL-CCA-X}}(b,k)$ is defined as:

$$\begin{array}{l} Expt_{\mathcal{A}}^{\texttt{CL-CCA-X}}(b,k): \\ (mpk, msk) \xleftarrow{\$} \texttt{Setup}(1^k) \\ (m_0, m_1, \mathsf{ID}^*, state) \xleftarrow{\$} \mathcal{A}_1(1^k, mpk, aux) \\ C^* \xleftarrow{\$} \texttt{Encrypt}(m_b, pk_{\mathsf{ID}^*}, \mathsf{ID}^*, mpk) \\ b' \xleftarrow{\$} \mathcal{A}_2(C^*, state) \\ \text{Output } b' \end{array}$$

We insist that $\mathcal{A}_1$ outputs messages (m_0, m_1) such that $|m_0| = |m_1|$. The Type I security model ($\texttt{X} = \texttt{I}$) and the Type II security model ($\texttt{X} = \texttt{II}$) are distinguished by the value aux and the oracles to which the attacker has access. The Type I model is meant to represent an outside attacker and so $aux = \emptyset$. The Type II

model captures the actions of an honest-but-curious KGC and so $aux = msk$. We consider the following oracles:

- **Request public key:** the attacker supplies an identity ID and the oracle returns the public key pk_{ID} for that identity. If pk_{ID} has not previously been defined, the oracle generates it.
- **Replace public key:** the attacker supplies an identity ID and a public key $pk_{\mathsf{ID}} \in \mathcal{PK}$, and the oracle replaces any previously generated public key for ID with pk_{ID}. Such a query is only allowed for correctly shaped new keys. Recall that the model of [1] requires the well-formedness of pk_{ID} (and the existence of a secret value) to be publicly checkable.
- **Extract partial private key:** the attacker supplies an identity ID and the oracle returns the partial private key d_{ID} for that identity.
- **Extract private key:** the attacker supplies an identity ID and the oracle responds with the full private key sk_{ID} for that identity.
- **Strong decrypt (or decrypt):** the attacker supplies an identity ID and a ciphertext C, and the oracle responds by constructing a private key sk_{ID} that corresponds to the identity ID and its associated public key. The oracle returns the decryption of C under this private key. Note that the oracle has to respond to decryption oracle queries even if the public key for the identity has been replaced.

Definition 1. *A CLE scheme is Strong Type I secure if, for every PPT adversary $\mathcal{A}$ that respects the following oracle constraints*

- *$\mathcal{A}$ cannot extract the private key for the identity ID^* at any time,*
- *$\mathcal{A}$ cannot extract the private key of any identity for which it has replaced the public key,*
- *$\mathcal{A}$ cannot extract the partial private key of ID^* if $\mathcal{A}$ replaced the public key pk_{ID^*} before the challenge was issued,*
- *$\mathcal{A}_2$ cannot query the strong decrypt oracle on the challenge ciphertext C^* for the identity ID^* unless the public key pk_{ID^*} used to create the challenge ciphertext has been replaced,*

we have that $Adv_{\mathcal{A}}^{CL\text{-}CCA\text{-}I}(k)$ is negligible. In this model, $aux = \emptyset$.

Definition 2. *A CLE scheme is Strong Type II secure if, for every PPT adversary $\mathcal{A}$ that respects the following oracle constraints*

- *$\mathcal{A}$ cannot extract the private key for the identity ID^* at any time,*
- *$\mathcal{A}$ cannot extract the private key of any identity for which it has replaced the public key,*
- *$\mathcal{A}$ does not query the partial private key oracle (since it can compute them itself given msk),*
- *$\mathcal{A}_1$ cannot output a challenge identity ID^* for which it has replaced the public key,*
- *$\mathcal{A}_2$ cannot query the strong decrypt oracle on the challenge ciphertext C^* for the identity ID^* unless the public key pk_{ID^*} used to create the challenge ciphertext has been replaced.*

we have that $Adv_{\mathcal{A}}^{\text{CL-CCA-II}}(k)$ *is negligible. In the Type II model, we have* $aux = msk$*, i.e.* $\mathcal{A}_1$ *takes the master private key as an additional input.*

We note that the definition of Type II security only covers honest-but-curious KGCs, as originally defined by Al-Riyami and Paterson [1]. An alternative definition, proposed by Au *et al.* [2], attempts to model security against a KGC that can maliciously generate its master public and private keys. We note that our schemes are not secure in this model. Nevertheless, we claim that the original security model still captures a significant level of security and that the design of secure standard model schemes fitting the original definitions represents a significant step forward in the theory of certificateless encryption. We do not find it unrealistic to assume that KGCs are honest at key generation time and erase relevant crucial information in case they are later broken into. Furthermore, it is difficult to see how a scheme can be proven secure against malicious key generation centres and outside attackers in the standard model and with strong decryption oracles using known proof techniques. The recent work of Huang and Wong [21] proves the security of a scheme against malicious KGCs in the standard model but does not permit a Strong Type I adversary, so the construction of such a scheme should still be considered an open problem.

A certificateless encryption scheme is said to be strongly secure if it is both Strong Type I and Strong Type II secure. A certificateless encryption scheme is said to be passively secure if it is Strong Type I and Strong Type II secure against adversaries who make no decryption oracle queries.

3 Generic Construction

In this section we develop a generic construction of a strongly secure certificateless encryption scheme from a passively secure certificateless encryption scheme, a passively secure public key encryption scheme, and a non-interactive zero-knowledge proof system. We do this by adapting the ideas of Naor-Yung [27] and Sahai [29] to the certificateless setting. The requirement that the simulator be able to decrypt ciphertexts encrypted using arbitrary public keys makes the construction slightly more complicated than in the public-key encryption case.

We first recall the notion of an NP language and that of simulation-sound non-interactive zero-knowledge proof system. Our requirements are similar to those of Sahai [29], but slightly more demanding.

Definition 3. *A language* $L \in \{0,1\}^*$ *is an* NP *language (*$L \in$ NP*) if there exists a (deterministic) Turing machine* R *that is polynomial-time with respect to its first input and satisfies:*

$$x \in L \iff \exists w \in \{0,1\}^* \text{ such that } R(x,w) = 1$$

We require a NIZK proof system that is statistically sound, computationally simulation-sound and computationally zero-knowledge. We require statistical soundness because (at one point in the proof) we will be forced to simulate

a decryption oracle that can provide functionality that cannot be computed in polynomial-time, i.e. decrypting ciphertexts that are encrypted under adversarially chosen public keys.

Definition 4. *A statistically sound, computationally simulation-sound, and computationally zero knowledge non-interactive zero-knowledge proof system (NIZK) for a language* $L \in \mathrm{NP}$ *is a tuple* $\Pi = (f, P, V, S_1, S_2)$ *where* f *is a polynomial and* P, V, S_1 *and* S_2 *are probabilistic, polynomial-time Turing machines that satisfy the following conditions:*

- ***Complete:*** *For all* $x \in L$ *and all* w *such that* $R(x, w) = 1$*, and for all strings* $\sigma \in \{0,1\}^{f(k)}$*, we have that* $V(x, \pi, \sigma) = 1$ *for all* $\pi \xleftarrow{\$} P(x, w, \sigma)$*.*
- ***Simulation complete:*** *For all* $x \in \{0,1\}^*$ *and all strings* $(\sigma, \kappa) \xleftarrow{\$} S_1(1^k)$*, we have that* $V(x, \pi, \sigma) = 1$ *for all* $\pi \xleftarrow{\$} S_2(x, \kappa)$*.* κ *can be thought of as a secret key that allows* S_2 *to produce false proofs.*
- ***Statistically sound****: Almost all common random strings* σ *should not allow any false theorem to be proven. In other words,*

$$Pr[\exists\, x \in \{0,1\}^* \setminus L \;\exists\, \pi \in \{0,1\}^* \text{ such that } V(x, \pi, \sigma) = 1]$$

is negligible as a function of the security parameter k *where the probability is taken over the choice of* $\sigma \xleftarrow{\$} \{0,1\}^{f(k)}$*.*
- ***Simulation sound:*** *For all non-uniform PPT adversaries* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ *we have that* $Adv_{\mathcal{A}}^{\mathtt{NZIK\text{-}SS}}(k) = Pr[Expt_{\mathcal{A}}^{SS}(k) = 1]$ *is negligible as a function of* k*, where*

$Expt_{\mathcal{A}}^{SS}(k)$:	*Output 1 if and only if:*
$(\sigma, \kappa) \xleftarrow{\$} \mathcal{S}_1(1^k)$	• $(x', \pi') \neq (x, \pi)$
$(x, state) \xleftarrow{\$} \mathcal{A}_1(1^k, \sigma)$	• $x' \notin L$
$\pi \xleftarrow{\$} S_2(x, \kappa)$	• $V(x', \pi', \sigma) = 1$
$(x', \pi') \xleftarrow{\$} \mathcal{A}_2(\pi, state)$	

- ***Zero knowledge:*** *For all non-uniform PPT adversaries* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ *we have that*

$$Adv_{\mathcal{B}}^{\mathtt{NIZK\text{-}ZK}}(k) = |Pr[Expt_{\mathcal{A}}(k) = 1] - Pr[Expt_{\mathcal{A}}^{S}(k) = 1]|$$

is negligible as a function of k*, where*

$Expt_{\mathcal{A}}(k)$:	$Expt_{\mathcal{A}}^{S}(k)$:
$\sigma \xleftarrow{\$} \{0,1\}^{f(k)}$	$(\sigma, \kappa) \xleftarrow{\$} \mathcal{S}_1(1^k)$
$(x, w, state) \xleftarrow{\$} \mathcal{A}_1(1^k, \sigma)$	$(x, w, state) \xleftarrow{\$} \mathcal{A}_1(1^k, \sigma)$
If $R(x, w) = 0$*, then* $\pi \leftarrow \emptyset$	*If* $R(x, w) = 0$*, then* $\pi \leftarrow \emptyset$
Otherwise $\pi \xleftarrow{\$} P(x, w, \sigma)$	*Otherwise* $\pi \xleftarrow{\$} S_2(x, \kappa)$
Return $\mathcal{A}_2(\pi, state)$	*Return* $\mathcal{A}_2(\pi, state)$

Sahai [29] uses a (single theorem) computationally sound and computationally zero-knowledge NIZK proof system to construct a (multiple theorem) computationally sound, computationally simulation-sound and computationally

zero-knowledge NIZK proof system. This construction assumes that one-way permutations exist. A brief examination of the proof verifies that we can construct a statistically sound, computationally simulation-sound NIZK proof system from a statistically sound NIZK proof system. Furthermore, it is not difficult to verify that statistically sound NIZK proof systems can be constructed for any NP language using the techniques of Feige, Lapidot and Shamir [16] under the assumption that certified trapdoor permutations exist. This condition is relaxed by Bellare and Yung [5] to require only that trapdoor permutations exist. Therefore we can construct suitably secure NIZK proof systems under the assumption that trapdoor permutations exist. Our construction will also make use of a passively-secure encryption scheme.

Definition 5. *A triple of PPT algorithms $(\mathcal{G}, \mathcal{E}, \mathcal{D})$ is an encryption scheme if (1) $\mathcal{G}$ takes as input a security parameter 1^k and outputs a public key pk and a private key sk; (2) $\mathcal{E}$ takes as input a message $m \in \mathcal{M}$ and a public key pk, and outputs a ciphertext $C \in \mathcal{C}$; and (3) $\mathcal{D}$ takes as input a ciphertext $C \in C$ and a private key sk, and outputs either a message $m \in \mathcal{M}$ or the error symbol $\perp$. This encryption scheme is said to be passively secure if the difference in probabilities*

$$Adv_{\mathcal{A}}^{\texttt{PKE-CPA}}(k) = |Pr[Expt_{\mathcal{A}}^{\texttt{PKE-CPA}}(0,k) = 1] - Pr[Expt_{\mathcal{A}}^{\texttt{PKE-CPA}}(1,k) = 1]|$$

is negligible for every probabilistic, polynomial-time attacker $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$. The experiment $Expt_{\mathcal{A}}^{\texttt{PKE-CPA}}(b,k)$ is defined as

$$\begin{array}{l} Expt_{\mathcal{A}}^{\texttt{PKE-CPA}}(b,k)\text{:} \\ (pk, sk) \stackrel{\$}{\leftarrow} \mathcal{G}(1^k) \\ (m_0, m_1, state) \stackrel{\$}{\leftarrow} \mathcal{A}_1(1^k, pk) \\ C^* \stackrel{\$}{\leftarrow} \mathcal{E}(m_b, pk) \\ Return\ \mathcal{A}_2(C^*, state) \end{array}$$

where we insist that $|m_0| = |m_1|$.

We construct a strongly secure CLE scheme from a passively secure one and *two* distinct instances of a public-key encryption scheme. We use the NIZK proof system to prove that these independently generated ciphertexts all encrypt the same message. Let $(\texttt{Setup}, \texttt{Extract}, \texttt{SetSec}, \texttt{SetPriv}, \texttt{SetPub}, \texttt{Encrypt}, \texttt{Decrypt})$ be a passively secure CLE scheme and $(\mathcal{G}, \mathcal{E}, \mathcal{D})$ be a passively secure public-key encryption scheme. Furthermore, let (f, P, V, S_1, S_2) be a statistically sound and computationally simulation-sound NIZK proof system for the language

$$\begin{array}{rl} L = \{(C_1, pk, \mathsf{ID}, mpk_1, C_2, mpk_2, C_3, mpk_3) \mid \exists\, (m, r_1, r_2, r_3) & \\ \text{such that } C_1 = \texttt{Encrypt}(m, pk, \mathsf{ID}, mpk_1; r_1) & \\ \wedge\, C_2 = \mathcal{E}(m, mpk_2; r_2) \wedge C_3 = \mathcal{E}(m, mpk_3; r_3)\} & \end{array}$$

Let $(\texttt{Setup}', \texttt{Extract}, \texttt{SetSec}, \texttt{SetPriv}, \texttt{SetPub}, \texttt{Encrypt}', \texttt{Decrypt}')$ be the certificateless encryption scheme derived from the passively secure scheme and the algorithms given in Figure 1. We assume that users' public key pk and identity

$\texttt{Setup}'(1^k)$:
$(mpk_1, msk_1) \xleftarrow{\$} \texttt{Setup}(1^k)$
$(mpk_2, msk_2) \xleftarrow{\$} \mathcal{G}(1^k)$
$(mpk_3, msk_3) \xleftarrow{\$} \mathcal{G}(1^k)$
$\sigma \xleftarrow{\$} \{0,1\}^{f(k)}$
$mpk' \leftarrow (mpk_1, mpk_2, mpk_3, \sigma)$
$msk' \leftarrow msk_1$
Output (mpk', msk')

$\texttt{Encrypt}'(m, pk, \mathsf{ID}, mpk')$:
$r_1, r_2, r_3 \xleftarrow{\$} \{0,1\}^{poly(k)}$
$C_1 \xleftarrow{\$} \texttt{Encrypt}(m, pk, \mathsf{ID}, mpk_1; r_1)$
$C_2 \xleftarrow{\$} \mathcal{E}(m, mpk_2; r_2)$
$C_3 \xleftarrow{\$} \mathcal{E}(m, mpk_3; r_3)$
$x \leftarrow (C_1, pk, \mathsf{ID}, mpk_1, C_2, mpk_2, C_3, mpk_3)$
$\pi \xleftarrow{\$} P(x, (m, r_1, r_2, r_3), \sigma)$
$C \leftarrow (C_1, C_2, C_3, \pi)$
Output C

$\texttt{Decrypt}'(C, sk, mpk')$:
$x \leftarrow (C_1, pk, \mathsf{ID}, mpk_1, C_2, mpk_2, C_3, mpk_3)$
If $V(x, \pi, \sigma) \neq 1$ then output $\perp$
Otherwise set $m \xleftarrow{\$} \texttt{Decrypt}(C_1, sk, mpk)$
Output m

Fig. 1. A construction for a strongly secure certificateless encryption scheme

ID are included in their full private key sk. We also assume (for simplicity and without loss of generality) that the random tapes used by each of the algorithms is of length $poly(k)$.

Theorem 1. *If*

- $(\texttt{Setup}, \texttt{Extract}, \texttt{SetSec}, \texttt{SetPriv}, \texttt{SetPub}, \texttt{Encrypt}, \texttt{Decrypt})$ *is a passively secure certificateless encryption scheme,*
- $(\mathcal{G}, \mathcal{E}, \mathcal{D})$ *is a passively secure public-key encryption scheme,*
- (f, P, V, S_1, S_2) *is a statistically sound, computationally simulation-sound and computationally zero-knowledge NIZK proof system for the* NP *language*

$$L = \{(C_1, pk, \mathsf{ID}, mpk_1, C_2, mpk_2, C_3, mpk_3) \mid \exists\, (m, r_1, r_2, r_3) \text{ such that } C_1 = \texttt{Encrypt}(m, pk, \mathsf{ID}, mpk_1; r_1) \wedge C_2 = \mathcal{E}(m, mpk_2; r_2) \wedge C_3 = \mathcal{E}(m, mpk_3; r_3)\}$$

then the certificateless encryption scheme given in Figure 1 is secure in the Strong Type I and Strong Type II models.

The proof is given in the full version of the paper [14]. It depends upon the fact that the master private key msk' does not contain the decryption keys for the public-key encryption schemes (msk_2, msk_3) or the simulation key κ for the NIZK proof system. We stress that this proof only works against Strong Type II adversaries who follow the setup procedure precisely, including the secure deletion of (msk_2, msk_3) and κ. The scheme can be trivially broken by a KGC that can generate the master public key in an adversarial way. In the standard model, it remains an open problem to construct a scheme that is strongly secure against adversaries who can generate the master public key.

Remark 1. This construction can also be thought of as using a NIZK proof to bind the encryption of a message under a passively secure certificateless encryption scheme to the encryption of the same message under an IND-CCA2 secure encryption scheme. In the specific case of the construction that we have proposed, the IND-CCA2 encryption scheme is the Sahai [29] construction of an IND-CCA2 encryption scheme from two passively secure encryption schemes and a (separate) NIZK proof system. The proofs of security can easily be adapted to the case where an arbitrary IND-CCA2 secure encryption scheme is used.

Remark 2. We note that we may construct passively secure encryption schemes and suitably secure NIZK proof systems for any NP language from trapdoor one-way permutations [29]. Furthermore, we may construct passively secure CLE schemes from passively secure public-key encryption schemes and passively secure identity-based encryption schemes [25]. Hence, we can conclude that strongly secure certificateless encryption schemes exist provided that NIZK proof systems and passively secure identity-based encryption schemes exist. It is an open problem to show that a passively secure identity-based encryption scheme can be constructed from any recognised minimal assumption. Since it is possible to construct NIZK proof systems [10] and passively secure identity-based encryption schemes [30] under the DBDH assumption, we can conclude that there exists a strongly secure certificateless encryption schemes under the DBDH assumption alone.

Remark 3. Two public-key encryption scheme are required in order to provide security against attackers with access to a strong decryption oracle. In weaker security models, where the attacker does not have access to a strong decryption oracle, a single public-key encryption scheme suffices.

4 Concrete Construction

Our concrete construction for CLE uses *bilinear map groups*, i.e. groups $(\mathbb{G}, \mathbb{G}_T)$ of prime order p for which there is an efficiently computable mapping $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ with the following properties:

1. bilinearity: $e(g^a, h^b) = e(g, h)^{ab}$ for any $(g, h) \in \mathbb{G} \times \mathbb{G}$ and $a, b \in \mathbb{Z}$;
2. non-degeneracy: $e(g, h) \neq 1_{\mathbb{G}_T}$ whenever $g, h \neq 1_{\mathbb{G}}$.

In such groups, we require the intractability of the following decisional problem that was suggested for the first time in [7] as a natural variant of the DBDH and DDH problems.

Definition 6. *The* Decision **3**-Party Diffie-Hellman Problem *(3-DDH) is to decide if $T = g^{abc}$ given $(g^a, g^b, g^c, T) \in \mathbb{G}^4$. Formally, we define the advantage of a PPT algorithm $\mathcal{A}$ as*

$$Adv_{\mathcal{A}}^{3\text{-}DDH}(k) = \begin{array}{l} |Pr[1 \stackrel{\$}{\leftarrow} \mathcal{A}(g^a, g^b, g^c, T) \mid T \stackrel{\$}{\leftarrow} g^{abc} \ \wedge \ a, b, c \stackrel{\$}{\leftarrow} \mathbb{Z}_p^*] \\ \quad -Pr[1 \stackrel{\$}{\leftarrow} \mathcal{A}(g^a, g^b, g^c, T) \mid T \stackrel{\$}{\leftarrow} \mathbb{G} \ \wedge \ a, b, c \stackrel{\$}{\leftarrow} \mathbb{Z}_p^*]| \end{array}$$

We will assume that $Adv_{\mathcal{A}}^{3\text{-}DDH}(k)$ is negligible for all PPT algorithms $\mathcal{A}$.

Our scheme is easily adapted to work in the more general setting of prime-order groups $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ with a pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ (instantiable from ordinary elliptic curve unlike the symmetric configuration that requires supersingular curves), in which case we need to use the obvious variant of the above hardness assumption. We also require a hash function H drawn from a family of collision resistant hash functions.

Definition 7. *A hash function* $H \xleftarrow{\$} \mathcal{H}(k)$ *is* collision resistant *if for all PPT algorithms* $\mathcal{A}$ *the advantage*

$$Adv^{CR}_{\mathcal{A}}(k) = Pr[H(x) = H(y) \wedge x \neq y \mid (x, y) \xleftarrow{\$} \mathcal{A}(1^k, H) \wedge H \xleftarrow{\$} \mathcal{H}(k)]$$

is negligible as a function of the security parameter.

Our scheme is an extension of the chosen-ciphertext secure IBE obtained by applying ideas from Boyen, Mei and Waters [9] to the 2-level hierarchical extension of the Waters IBE.

$\texttt{Setup}(1^k, n)$**:** Let $(\mathbb{G}, \mathbb{G}_T)$ be bilinear map groups of order $p > 2^k$ and let g be a generator for $\mathbb{G}$. Set $g_1 = g^\gamma$, for a random $\gamma \xleftarrow{\$} \mathbb{Z}_p^*$, and pick a group element $g_2 \xleftarrow{\$} \mathbb{G}$ and vectors $(u', u_1, \ldots, u_n)$, $(v', v_1, \ldots, v_n) \xleftarrow{\$} \mathbb{G}^{n+1}$. We note that these vectors define the hash functions

$$F_u(\mathsf{ID}) = u' \prod_{i=1}^{n} u_j^{i_j} \qquad \text{and} \qquad F_v(w) = v' \prod_{i=1}^{n} v_j^{w_j}$$

where $\mathsf{ID} = i_1 i_2 \ldots i_n$ and $w = w_1 w_2 \ldots w_n$. We also select a collision-resistant hash function $H : \{0,1\}^* \to \{0,1\}^n$. The master public key is

$$mpk \leftarrow (g, g_1, g_2, u', u_1, \ldots, u_n, v', v_1, \ldots, v_n)$$

and the master secret[1] is $msk \leftarrow g_2^\gamma$.

$\texttt{Extract}(mpk, \gamma, \mathsf{ID})$**:** Pick $r \xleftarrow{\$} \mathbb{Z}_p^*$ and return $d_{\mathsf{ID}} \leftarrow (d_1, d_2) = (g_2^\gamma \cdot F_u(\mathsf{ID})^r, g^r)$.

$\texttt{SetSec}(mpk)$**:** Return a randomly chosen secret value $x_{\mathsf{ID}} \xleftarrow{\$} \mathbb{Z}_p^*$.

$\texttt{SetPub}(x_{\mathsf{ID}}, mpk)$**:** Return $pk_{\mathsf{ID}} \leftarrow (X, Y) = (g^{x_{\mathsf{ID}}}, g_1^{x_{\mathsf{ID}}})$.

$\texttt{SetPriv}(x_{\mathsf{ID}}, d_{\mathsf{ID}}, mpk)$**:** Parse d_{ID} into (d_1, d_2), choose $r' \xleftarrow{\$} \mathbb{Z}_p^*$ and set the private key to

$$sk_{\mathsf{ID}} \leftarrow (s_1, s_2) = (d_1^{x_{\mathsf{ID}}} \cdot F_u(\mathsf{ID})^{r'}, d_2^{x_{\mathsf{ID}}} \cdot g^{r'}) = (g_2^{\gamma x_{\mathsf{ID}}} \cdot F_u(\mathsf{ID})^t, g^t)$$

with $t = r x_{\mathsf{ID}} + r'$.

$\texttt{Encrypt}(m, pk_{\mathsf{ID}}, \mathsf{ID}, mpk)$**:** To encrypt $m \in \mathbb{G}_T$, parse $\mathsf{pk}_{\mathsf{ID}}$ as (X, Y), then check that it has the right shape (i.e. that $e(X, g_1)/e(g, Y) = 1_{\mathbb{G}_T}$). If so, choose $s \xleftarrow{\$} \mathbb{Z}_p^*$ and compute

$$C = (C_0, C_1, C_2, C_3) \leftarrow \left(m \cdot e(Y, g_2)^s, g^s, F_u(\mathsf{ID})^s, F_v(w)^s\right)$$

where $w \leftarrow H(C_0, C_1, C_2, \mathsf{ID}, pk_{\mathsf{ID}})$.

[1] In order to ensure security against Type II attacks according to definition 2, the discrete logarithms of elements g_2, $u', u_1, \ldots, u_n, v', v_1, \ldots, v_n$ w.r.t. the base g are not part of the master secret and should be deleted after key generation by the KGC.

$\texttt{Decrypt}(C, sk_{\mathsf{ID}}, mpk)$**:** Parse C as (C_0, C_1, C_2, C_3) and the private key sk_{ID} as (s_1, s_2). Check that

$$e\big(C_1, F_u(\mathsf{ID}) \cdot F_v(w)\big) = e\big(g, C_2 \cdot C_3\big)$$

where $w \leftarrow H(C_0, C_1, C_2, \mathsf{ID}, pk_{\mathsf{ID}})$, and reject C if those conditions do not hold. Otherwise, return

$$m \leftarrow C_0 \cdot \frac{e(C_2, s_2)}{e(C_1, s_1)}$$

To check the completeness, we note that private keys (s_1, s_2) satisfy

$$e(g, s_1) = e(Y, g_2) \cdot e(F_u(\mathsf{ID}), s_2) \qquad \text{and so} \qquad e(C_1, s_1) = e(Y, g_2)^s \cdot e(C_2, s_2)\,.$$

To speed up the decryption algorithm using ideas from [23], we observe that the receiver can randomly choose $\alpha \stackrel{\$}{\leftarrow} \mathbb{Z}_p^*$ and directly return

$$m = C_0 \cdot \frac{e(C_2, s_2 \cdot g^\alpha) \cdot e(C_3, g^\alpha)}{e(C_1, s_1 \cdot F_u(\mathsf{ID})^\alpha \cdot F_v(w)^\alpha)}$$

which is the actual plaintext if C was properly encrypted and a random element of $\mathbb{G}_T$ otherwise. The well-formedness of C is thus implicitly checked and a product of three pairings suffices to decipher the message. This is sufficient to satisfy our security models; however, it should be noted that this system has the disadvantage of outputting a random message when presented with an invalid ciphertext. This may be a problem in some applications. In the same way, the public key validation can be made implicit at encryption: given $pk_{\mathsf{ID}} = (X, Y)$, the sender picks $\beta \stackrel{\$}{\leftarrow} \mathbb{Z}_p^*$ and computes $C_0 = m \cdot e(Y, g_2^s \cdot g^{s\beta}) / e(X, g_1^{s\beta})$ which actually encrypts m whenever pk_{ID} has the correct shape and results in an invalid ciphertext otherwise.

We have the following security results for this concrete scheme:

Theorem 2. *Suppose $\mathcal{A}$ is a Strong Type I adversary that runs in time t, makes at most q_d decryption queries, q_{ppk} partial private key queries, and q_{pk} private key queries. Then there exists*

- *an adversary $\mathcal{A}'$ against the 3-DDH problem that has advantage $Adv_{\mathcal{A}'}^{3\text{-}DDH}(k)$ and runs in time $O(t) + O(\epsilon^{-2} \ln \delta^{-1})$ for sufficiently small ϵ and δ, and*
- *an adversary $\mathcal{A}''$ against the collision resistance of the hash function H that runs in time $O(t)$ and has advantage $Adv_{\mathcal{A}''}^{CR}(k)$*

such that the advantage of $\mathcal{A}$ is bounded by

$$Adv_{\mathcal{A}}^{CL\text{-}CCA\text{-}I}(k) < 8(q_{ppk} + q_{pk}) q_d (n+1)^2 \cdot (8 \cdot Adv_{\mathcal{A}'}^{3\text{-}DDH}(k) + \delta) + Adv_{\mathcal{A}''}^{CR}(k)\,.$$

The proof of this theorem is given in the full version of the paper [14]; it uses ideas from [9,30]. Namely, the mapping F_v is chosen so as to have $F_v(w) = g_2^{J_v(w)} g^{K_v(w)}$, for certain functions J_v and K_v, in the simulation of the attack

environment. Hence, for any valid ciphertext $C = (C_0, C_1, C_2, C_3)$, we have $C_1 = g^s$ and $C_3 = F_v(w)^s$, for some $s \in \mathbb{Z}_p^*$, and the simulator can extract

$$g_2^s = (C_3 / C_1^{K_v(w)})^{1/J_v(w)}$$

whenever $J_v(w) \neq 0 \bmod p$. Hence, the simulator can compute $e(Y, g_2)^s$ regardless of whether the public key $pk = (X, Y)$ was replaced or not.

Theorem 3. *Suppose $\mathcal{A}$ is a Strong Type II adversary that runs in time t and makes at most q_d decryption queries and q_{pk} private key queries. Then there exists*

- *an adversary $\mathcal{A}'$ against the 3-DDH problem that has advantage $Adv_{\mathcal{A}'}^{3\text{-}DDH}(k)$ and runs in time $O(t) + O(\epsilon^{-2} \ln \delta^{-1})$ for sufficiently small ϵ and δ, and*
- *an adversary $\mathcal{A}''$ against the collision resistance of the hash function H that runs in time $O(t)$ and has advantage $Adv_{\mathcal{A}''}^{CR}(k)$*

such that the advantage of $\mathcal{A}$ is bounded by

$$Adv_{\mathcal{A}}^{CL\text{-}CCA\text{-}II}(k) < 8 q_{pk} q_d (n+1)^2 \cdot (8 \cdot Adv_{\mathcal{A}'}^{3\text{-}DDH}(k) + \delta) + Adv_{\mathcal{A}''}^{CR}(k)\,.$$

The proof of this theorem is given in the full version of the paper [14] and uses similar ideas to the proof of Theorem 2.

The reductions given in the proofs of Theorems 2 and 3 leave definite room for improvement since chosen-ciphertext security is achieved by applying the Boyen-Mei-Waters techniques [9] to a 2-level HIBE.

One solution to improve the reduction is to use the Canetti-Halevi-Katz [11] or Boneh-Katz [8] techniques that significantly lengthen ciphertexts and/or introduce additional assumptions for the security of the scheme. If we borrow ideas from [34] and generate the checksum value $C_3 = F(w)^s$ using a chameleon hash function [24] in instead of Waters' "hash", an interesting tradeoff can be achieved. In the above variant, a single element of $\mathbb{Z}_p^*$ (acting as random coins used to compute of the chameleon hash function) should be appended to ciphertexts and the degradation factor q_d is avoided in both reductions. Using a chameleon hash function built upon Pedersen's discrete-logarithm-based trapdoor commitment [28], the resulting combination does not imply any additional intractability assumption for the security of the final scheme.

Acknowledgements

The authors would like to thank Douglas Wikström for an initial conversation about whether it would be possible to construct strong certificateless encryption using Naor-Yung style techniques, and Eike Kiltz for several discussions on artificial aborts. The authors would also like to thank the PKC referees and David Galindo for their helpful comments. The second author acknowledges the Belgian National Fund for Scientific Research (F.R.S.-F.N.R.S.) for their support. This work was supported in part by the European Commission under Contract IST-2002-507932 ECRYPT.

References

1. Al-Riyami, S.S., Paterson, K.G.: Certificateless public key cryptography. In: Laih, C.-S. (ed.) ASIACRYPT 2003. LNCS, vol. 2894, pp. 452–473. Springer, Heidelberg (2003)
2. Au, M.H., Chen, J., Liu, J.K., Mu, Y., Wong, D.S., Yang, G.: Malicious KGC attack in certificateless cryptography. In: Proc. ACM Symposium on Information, Computer and Communications Security, ACM Press, New York (2007)
3. Baek, J., Safavi-Naini, R., Susilo, W.: Certificateless public key encryption without pairing. In: Zhou, J., López, J., Deng, R.H., Bao, F. (eds.) ISC 2005. LNCS, vol. 3650, pp. 134–148. Springer, Heidelberg (2005)
4. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: Proc. of the First ACM Conference on Computer and Communications Security, pp. 62–73 (1993)
5. Bellare, M., Yung, M.: Certifying permutations: Non-interactive zero-knowledge based on any trapdoor permutation. Journal of Cryptology 9(1), 149–166 (1996)
6. Bentahar, K., Farshim, P., Malone-Lee, J., Smart, N.P.: Generic constructions of identity-based and certificateless KEMs (2005), `http://eprint.iacr.org/2005/058`
7. Boneh, D., Franklin, M.: Identity based encryption from the Weil pairing. SIAM J. of Computing 32(3), 586–615 (2003)
8. Boneh, D., Katz, J.: Improved efficiency for CCA-secure cryptosystems built using identity-based encryption. In: Menezes, A. (ed.) CT-RSA 2005. LNCS, vol. 3376, pp. 87–103. Springer, Heidelberg (2005)
9. Boyen, X., Mei, Q., Waters, B.: Direct chosen ciphertext security from identity-based techniques. In: Proc. of the 12th ACM Conference on Computer and Communications Security, pp. 320–329 (2005)
10. Canetti, R., Halevi, S., Katz, J.: A forward-secure public-key encryption scheme. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 255–271. Springer, Heidelberg (2003)
11. Canetti, R., Halevi, S., Katz, J.: Chosen-ciphertext security from identity-based encryption. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 207–222. Springer, Heidelberg (2004)
12. Cheng, Z., Comley, R.: Efficient certificateless public key encryption (2005), `http://eprint.iacr.org/2005/012/`
13. Dent, A.W.: A survey of certificateless encryption schemes and security models (2006), `http://eprint.iacr.org/2006/211`
14. Dent, A.W., Libert, B., Paterson, K.G.: Certificateless encryption schemes strongly secure in the standard model (2007), `http://eprint.iacr.org/2007/121`
15. Dodis, Y., Katz, J.: Chosen-ciphertext security of multiple encryption. In: Kilian, J. (ed.) TCC 2005. LNCS, vol. 3378, pp. 188–209. Springer, Heidelberg (2005)
16. Feige, U., Lapidot, D., Shamir, A.: Multiple noninteractive zero knowledge proofs under general assumptions. SAIM Journal on Computing 29(1), 1–28 (1999)
17. Fujisaki, E., Okamoto, T.: How to enhance the security of public-key encryption at minimal cost. In: Imai, H., Zheng, Y. (eds.) PKC 1999. LNCS, vol. 1560, pp. 53–68. Springer, Heidelberg (1999)
18. Galindo, D., Morillo, P., Ràfols, C.: Breaking Yum and Lee generic constructions of certificate-less and certificate-based encryption schemes. In: Atzeni, A.S., Lioy, A. (eds.) EuroPKI 2006. LNCS, vol. 4043, pp. 81–91. Springer, Heidelberg (2006)

19. Gentry, C.: Certificate-based encryption and the certificate revocation problem. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 272–293. Springer, Heidelberg (2003)
20. Hu, B.C., Wong, D.S., Zhang, Z., Deng, X.: Key replacement attack against a generic construction of certificateless signature. In: Batten, L.M., Safavi-Naini, R. (eds.) ACISP 2006. LNCS, vol. 4058, pp. 235–246. Springer, Heidelberg (2006)
21. Huang, Q., Wong, D.S.: Generic certificateless encryption in the standard model. In: Miyaji, A., Kikuchi, H., Rannenberg, K. (eds.) IWSEC 2007. LNCS, vol. 4752, pp. 278–291. Springer, Heidelberg (2007)
22. Huang, X., Susilo, W., Mu, Y., Zhang, F.: On the security of certificateless signature schemes from Asiacrypt 2003. In: Desmedt, Y.G., Wang, H., Mu, Y., Li, Y. (eds.) CANS 2005. LNCS, vol. 3810, pp. 13–25. Springer, Heidelberg (2005)
23. Kiltz, E., Galindo, D.: Direct chosen-ciphertext secure identity-based key encapsulation without random oracles. In: Batten, L.M., Safavi-Naini, R. (eds.) ACISP 2006. LNCS, vol. 4058, pp. 336–347. Springer, Heidelberg (2006)
24. Krawczyk, H., Rabin, T.: Chameleon signatures. In: the Proceedings of the Network and Distributed Systems Symposium (NDSS 2000), pp. 143–154 (2000)
25. Libert, B., Quisquater, J.-J.: On constructing certificateless cryptosystems from identity based encryption. In: Yung, M., Dodis, Y., Kiayias, A., Malkin, T.G. (eds.) PKC 2006. LNCS, vol. 3958, pp. 474–490. Springer, Heidelberg (2006)
26. Liu, J.K., Au, M.H., Susilo, W.: Self-generated-certificate public key cryptography and certificateless signature/encryption scheme in the standard model. In: Proc. ACM Symposium on Information, Computer and Communications Security, ACM Press, New York (2007)
27. Naor, M., Yung, M.: Public-key cryptosystems provably secure against chosen ciphertext attacks. In: Proc. 22nd Symposium on the Theory of Computing, STOC 1990, pp. 427–437. ACM Press, New York (1990)
28. Pedersen, T.: Non-interactive and information-theoretic secure verifiable secret sharing. In: Feigenbaum, J. (ed.) CRYPTO 1991. LNCS, vol. 576, pp. 129–140. Springer, Heidelberg (1992)
29. Sahai, A.: Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In: 40th Annual Symposium on Foundations of Computer Science, FOCS 1999, pp. 543–553. IEEE Computer Society Press, Los Alamitos (1999)
30. Waters, B.: Efficient identity-based encryption without random oracles. In: Cramer, R.J.F. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 114–127. Springer, Heidelberg (2005)
31. Yum, D.H., Lee, P.J.: Generic construction of certificateless encryption. In: Laganà, A., Gavrilova, M.L., Kumar, V., Mun, Y., Tan, C.J.K., Gervasi, O. (eds.) ICCSA 2004. LNCS, vol. 3043, pp. 802–811. Springer, Heidelberg (2004)
32. Yum, D.H., Lee, P.J.: Generic construction of certificateless signature. In: Wang, H., Pieprzyk, J., Varadharajan, V. (eds.) ACISP 2004. LNCS, vol. 3108, pp. 200–211. Springer, Heidelberg (2004)
33. Yum, D.H., Lee, P.J.: Identity-based cryptography in public key management. In: Katsikas, S.K., Gritzalis, S., Lopez, J. (eds.) EuroPKI 2004. LNCS, vol. 3093, pp. 71–84. Springer, Heidelberg (2004)
34. Zhang, R.: Tweaking TBE/IBE to PKE transforms with chameleon hash functions. In: Katz, J., Yung, M. (eds.) ACNS 2007. LNCS, vol. 4521, pp. 323–339. Springer, Heidelberg (2007)
35. Zhang, Z., Wong, D.S., Xu, J., Feng, D.: Certificateless public-key signature: Security model and efficient construction. In: Zhou, J., Yung, M., Bao, F. (eds.) ACNS 2006. LNCS, vol. 3989, pp. 293–308. Springer, Heidelberg (2006)