

Binary Tree Encryption: Constructions and Applications

JONATHAN KATZ*

Department of Computer Science
University of Maryland
College Park, MD, USA
jkatz@cs.umd.edu

Abstract. *Binary tree encryption* (BTE), a relaxation of hierarchical identity-based encryption (HIBE), has recently emerged as a useful and intriguing primitive. On the one hand, the definition of security for BTE is sufficiently “weak” that — in contrast to HIBE — constructions of BTE *in the standard model* are known. On the other hand, BTE is sufficiently powerful that it yields a number of applications which are important from both a theoretical and a practical point of view.

This survey presents the basic definitions of BTE and also highlights some recent applications of BTE to forward-secure encryption, identity-based and hierarchical identity-based encryption, chosen-ciphertext security, and adaptively-secure encryption.

1 Introduction

The notion of identity-based cryptography has long fascinated researchers [23]. Loosely speaking, in such a scheme *any* identity (i.e., bit-string) can serve as a public key. In somewhat more detail, there is a (trusted) private-key generator PKG who generates master system parameters \mathbf{params} along with a master secret key sk . For any identity $id \in \{0, 1\}^*$ the PKG can use sk to compute a secret key SK_{id} corresponding to this identity. The pair (id, SK_{id}) then functions as a standard public-/private-key pair (with the important distinction that id can be any string!) whose functionality is determined by the underlying identity-based scheme. (The PKG would presumably authenticate the identity of the person claiming “ id ” before giving them the corresponding secret key SK_{id} . However, this is outside the scope of the present discussion.) An identity-based system is secure (informally) if knowledge of the secret keys corresponding to any arbitrary-size set of identities $\mathcal{I} = \{id_1, \dots, id_n\}$ does not allow an adversary to “break” the scheme (in the appropriate sense) for any $id' \notin \mathcal{I}$.

Shamir [23] was the first to suggest an implementation of an identity-based signature scheme. Following this, many provably-secure proposals for identity-based signature and identification schemes followed (e.g., [13, 16]); some of these

* Portions of this work were supported by NSF grant #ANI-0310751.

constructions were recently generalized and expanded upon in [11]. Although these constructions are proven secure in the random oracle model, note that it is also possible to construct identity-based signatures in the standard model based on any “regular” signature scheme (see [11]).

Recently, Boneh and Franklin [5] and Cocks [10] resolved a long-standing open problem by constructing the first identity-based public-key *encryption* schemes. Both of these constructions are proven secure in the random oracle model. Since encryption schemes are the focus of this article (and are more interesting in the sense that they are more difficult to construct), we consider only encryption from now on.

It is natural to extend the notion of identity-based encryption (IBE) to include *hierarchical* identity-based encryption (HIBE). In an HIBE scheme, the PKG (as above) issues secret keys to “first-level” identities $id \in \{0, 1\}^*$; furthermore, anyone knowing the secret key SK_{id_1} corresponding to a “first-level” identity id_1 can issue a secret key $SK_{id_1||id_2}$ corresponding to any “second-level” identity $id_1||id_2$ (for arbitrary $id_2 \in \{0, 1\}^*$). More generally, let $ID = (id_1||\dots||id_t)$ and let SK_{ID} be the secret key corresponding to this identity. Then for any string $id_{t+1} \in \{0, 1\}^*$ and identity $ID' \stackrel{\text{def}}{=} (ID||id_{t+1})$, knowledge of SK_{ID} enables computation of a key $SK_{ID'}$. As before, in all these cases the pair (ID, SK_{ID}) functions as a “standard” public-/private-key pair. The security requirement is modified in the obvious way: now, one requires that knowledge of the secret keys corresponding to any arbitrary-size set of identities $\mathcal{I} = \{ID_1, \dots, ID_n\}$ should not enable an adversary to “break” the scheme (in some appropriate sense) for any ID' having no ancestors in \mathcal{I} , where the *ancestors* of an identity $ID = (id_1||\dots||id_n)$ are all identities of the form $(id_1||\dots||id_i)$ for $i \leq n$.

Horwitz and Lynn [17] were the first to suggest the notion of HIBE, and they also propose a partial solution handling identities of depth two. Gentry and Silverberg [14] were the first to give a complete solution to this problem, and they construct and prove secure a scheme supporting identities of arbitrary (constant) depth. Both of these constructions build on the IBE scheme of Boneh and Franklin [5], and both are proven secure in the random oracle model.

1.1 Binary Tree Encryption

It can be immediately noticed that the identities in a hierarchical identity-based scheme correspond in the natural way to nodes in a tree. Specifically, one may associate the PKG with the root of the tree, the “first-level” identities with the nodes of depth one (i.e., the children of the root), and the identity $ID' = (id_1||\dots||id_{t+1})$ with a node at depth $t+1$ which is the child of a node at depth t which is in turn associated with $ID = (id_1||\dots||id_t)$.

In a scheme as outlined above, the identity hierarchy yields a tree of *unbounded* degree. In contrast, a *binary* tree encryption (BTE) scheme [7] — as the name suggests — considers only an identity hierarchy in the form of a *binary* tree (i.e., a tree in which each node has degree two). Viewing BTE as a conceptual relaxation of HIBE, one obtains a scheme in which the PKG may

potentially issue secret keys to (only) two “identities”: 0 and 1. In turn, the “identity” 0 (with knowledge of the appropriate secret key SK_0) can potentially issue secret keys for the “identities” 00 and 01; an analogous statement holds for the “identity” 1. More generally (and dispensing with the purely imaginary concept of “identities” here), the secret key SK_w corresponding to the binary string $w \in \{0, 1\}^t$ enables derivation of the secret keys SK_{w0} and SK_{w1} corresponding to the strings $w0, w1 \in \{0, 1\}^{t+1}$. As in the case of hierarchical identity-based encryption, each pair (w, SK_w) functions as a public-/private-key pair. A definition of security is developed in a way similar (but slightly different) to that discussed above in the context of hierarchical identity-based encryption; a formal definition appears in Section 2.

“Relaxing” the notion of hierarchical identity-based encryption in this way turns out to be an incredibly powerful idea. For one, a BTE scheme supporting trees of arbitrary *polynomial* depth has recently been constructed and proven secure *in the standard model* [7] (recall that in the case of HIBE, only a scheme of *constant* depth with a proof of security in the *random oracle model* [14] is known). The proof relies on a reasonable number-theoretic assumption (namely, the *decisional bilinear Diffie-Hellman assumption*) related¹ to that used by Boneh and Franklin in constructing their ID-based scheme [5]. This construction of a BTE scheme builds on the construction of Gentry and Silverberg with an important “twist”: because a *binary* tree is used (and because of a slight relaxation of the security definition), it is possible to replace the random oracle with a $\text{poly}(k)$ -wise independent hash function, a description of which is included as part of the master system parameters.

Equally important, the “relaxed” notion of BTE is surprisingly powerful and suffices for a number of applications:

- BTE was used to construct the first *forward-secure encryption scheme* [7] (indeed, the notion of BTE was introduced in the context of research directed toward solving this problem). Note that this is currently the only methodology known for achieving forward-secure encryption.
- BTE implies both identity-based encryption as well as hierarchical identity-based encryption [7], albeit only with respect to a *non-adaptive* definition of security which is weaker than the definition originally proposed [5]. This results in the first constructions of IBE and HIBE schemes which may be proven secure in the standard model.
- Recent work [8] shows that any IBE scheme (even if “only” secure against non-adaptive attacks) can be used to construct a standard public-key encryption scheme secure against adaptive chosen-ciphertext attacks (i.e., a CCA-secure scheme; cf. [3]). Given the result mentioned above, this yields a new construction of a CCA-secure encryption scheme in the standard model. Interestingly, the construction seems not to follow the paradigms underlying all previous constructions of CCA-secure encryption schemes (cf. [12]).

¹ It is also possible to base a BTE scheme on the identical assumption used by Boneh and Franklin (in the standard model) at the expense of a loss in efficiency.

- Finally, it has recently been shown [9] how to construct an adaptively-secure encryption scheme with “short” keys (namely, with keys shorter than the length of all plaintext messages sent — in fact, the length of plaintext to be encrypted may be *a priori* unbounded) based on any forward-secure encryption scheme plus an NIZK proof system.² We comment that adaptively-secure encryption with “short” keys is impossible [21] unless some form of key-evolving techniques (such as those used in forward-secure encryption schemes) are used.

It is hoped that the above results represent just the “tip of the iceberg” and that further applications of BTE will be developed.

1.2 Outline

The remainder of the paper is organized as follows. In Section 2, we give a formal definition of binary tree encryption as well as the corresponding definition of security. In Section 3, we state the known results regarding constructions of BTE. The applications of BTE, as highlighted above, are discussed in Section 4. The treatment given here is at a relatively high level; the interested reader is referred to the original papers [7–9] for additional information.

2 Definitions

Definitions related to identity-based encryption [5] and hierarchical identity-based encryption [14] are given elsewhere; for the purposes of understanding the definition of binary tree encryption, the informal descriptions provided in the Introduction should suffice. We thus begin with a formal definition of *binary tree encryption* (BTE), taken from [7]:

Definition 1. *A (public-key) binary tree encryption (BTE) scheme is a 4-tuple of PPT algorithms (Gen, Der, Enc, Dec) such that:*

- The key generation algorithm **Gen** takes as input a security parameter 1^k and a value ℓ for the depth of the tree. It returns a master public key PK and an initial (root) secret key SK_ε . (We assume that the values of k and ℓ are implicit in PK and all node secret keys.)
- The key derivation algorithm **Der** takes as input PK , the name of a node $w \in \{0, 1\}^{<\ell}$, and its secret key SK_w . It returns secret keys SK_{w0}, SK_{w1} for the two children of w .
- The encryption algorithm **Enc** takes as input PK , the name of a node $w \in \{0, 1\}^{\leq \ell}$, and a message M . It returns a ciphertext C .
- The decryption algorithm **Dec** takes as input PK , the name of a node $w \in \{0, 1\}^{\leq \ell}$, its secret key SK_w , and a ciphertext C . It returns a message M .

² Interestingly, it is shown in [7] how to construct an NIZK proof system based on the same number-theoretic assumption used for the forward-secure encryption scheme.

For correctness, we require that for any (PK, SK_ε) output by $\text{Gen}(1^k, \ell)$, any node $w \in \{0, 1\}^{\leq \ell}$ and secret key SK_w correctly generated for this node, and any message M , we have $M = \text{Dec}(PK, w, SK_w, \text{Enc}(PK, w, M))$.

The security notion for BTE is largely similar to the security notion for HIBE, with the key difference being that the present definition requires the attacker to commit to the node to be attacked (i.e., the “target node”) *in advance*, before seeing the public key and before asking any key exposure queries. This type of attack is called a *selective-node (SN) attack*. While the resulting definition is weaker than a definition which allows the adversary to *adaptively* select the target node, we stress again that this “weaker” definition suffices for all the applications mentioned herein. Furthermore, it is (in part) this weakening of the definition which allows for a construction of BTE in the standard model.

Definition 2. *A BTE scheme is secure against selective-node, chosen-plaintext attacks (SN-CPA) if for all polynomially-bounded functions $\ell(\cdot)$, the advantage of any PPT adversary A in the following game is negligible in the security parameter:*

1. $A(1^k, \ell(k))$ outputs a name $w^* \in \{0, 1\}^{\leq \ell(k)}$ of a node.
2. Algorithm $\text{Gen}(1^k, \ell(k))$ outputs (PK, SK_ε) . In addition, algorithm $\text{Der}(\cdot \cdot \cdot)$ is run to generate the secret keys of all the nodes on the path from the root to w^* (we denote this path by P), and also the secret keys for the two children of w^* (if $|w^*| < \ell$).
3. The adversary is given PK and also the secret keys $\{SK_w\}$ for all nodes w of the following form:
 - $w = w'\bar{b}$, where $w'b$ is a prefix of w^* and $b \in \{0, 1\}$ (i.e., w is a sibling of some node in P);
 - $w = w^*0$ or $w = w^*1$ (i.e., w is a child of w^* ; this is only when $|w^*| < \ell$).*(Note that this allows the adversary to compute $SK_{w'}$ for any node $w' \in \{0, 1\}^{\leq \ell(k)}$ that is not a prefix of w^* .)*
4. The adversary generates a request $\text{challenge}(M_0, M_1)$. A random bit b is selected and the adversary is given $C^* = \text{Enc}(PK, w^*, M_b)$.

At the end of the game the adversary outputs $b' \in \{0, 1\}$; it succeeds if $b' = b$. The adversary’s advantage is the absolute value of the difference between its success probability and $1/2$.

Security against chosen-ciphertext attacks (denoted SN-CCA) is defined as the obvious extension of the above; see [7] for details.

3 Constructions of Secure BTE Schemes

We limit ourselves to listing the known results regarding constructions of secure BTE schemes, and to a tabulation of their complexity (as a function of the tree depth); the reader is referred to [7] for further details. All constructions

mentioned below (indeed, all known constructions of BTE) rely on variants of the so-called *Bilinear Diffie-Hellman (BDH) assumption*. This assumption was first formally defined by Boneh and Franklin [5], motivated by earlier work of Joux [18] and Joux and Nguyen [19].

One of the main results of [7] is the following:

Theorem 1. *Assuming the decisional BDH assumption, there exists a BTE scheme secure in the sense of SN-CPA.*

It is easy to modify the construction so that it relies only on the (possibly weaker) *computational* BDH assumption (this may be done by using a hardcore predicate of the computational BDH problem, and encrypting bit-by-bit). However, this modification comes at the expense of a significant loss of efficiency.

Two generic techniques for achieving chosen-ciphertext security for an *arbitrary* BTE scheme have been proposed. The first [7] relies on non-malleable non-interactive zero-knowledge (NIZK) proofs, adapting an approach due to Naor and Yung [20] and Sahai [22] in the context of making “standard” public-key encryption schemes secure against chosen-ciphertext attacks. Interestingly, in the process of developing this solution it is also shown how non-malleable NIZK may be based on any publicly-verifiable trapdoor predicate (this notion, introduced by [11, 7], generalizes the notion of trapdoor permutations), and furthermore how the decisional BDH assumption naturally gives rise to such predicates. Putting this together gives the following result:

Theorem 2. *Assuming the decisional BDH assumption, there exists a BTE scheme secure in the sense of SN-CCA.*

Because the above relies on NIZK proofs of generic NP statements, it should properly be regarded as a feasibility result rather than as a method for constructing efficient schemes. Recently [8], a more efficient method for achieving chosen-ciphertext security for an arbitrary BTE scheme was proposed; this method (in particular) avoids any zero-knowledge proofs and instead relies on one-time signature schemes (which may be constructed from any BTE scheme). This gives an alternate proof of the above theorem, via a more practical construction.

The above results all hold in the standard model. If one is willing to assume the random oracle model, improved efficiency can be achieved. For one, it should be clear that any HIBE scheme which is secure for a non-adaptive choice of the target identity is also a BTE scheme; thus, the construction of [14] may be used. One way to view this is as simply replacing the $\text{poly}(k)$ -wise independent hash function in the construction of [7] by a random oracle (which, of course, is also a $\text{poly}(k)$ -wise independent hash function). This leads to improved efficiency since a $\text{poly}(k)$ -wise independent hash function is (relatively) expensive to generate and evaluate — in particular, requiring time $O(\text{poly}(k))$ — while for a random oracle these operations are all assumed to take time $O(1)$. Furthermore, essentially the same scheme (with but one additional call to the random oracle) may be based on the (possibly weaker) computational BDH assumption rather than

| | Standard model | Random oracle model |
|----------------------------|-----------------------------|---------------------|
| Master key generation time | $\mathcal{O}(\ell)$ | $\mathcal{O}(1)$ |
| Encryption/decryption time | $\tilde{\mathcal{O}}(\ell)$ | $\mathcal{O}(\ell)$ |
| Key derivation time | $\mathcal{O}(\ell)$ | $\mathcal{O}(1)$ |
| Ciphertext length | $\mathcal{O}(\ell)$ | $\mathcal{O}(\ell)$ |
| Public key size | $\mathcal{O}(\ell)$ | $\mathcal{O}(1)$ |
| Secret key size | $\mathcal{O}(\ell)$ | $\mathcal{O}(\ell)$ |

Table 1. Summary of dependencies on the depth of the tree ℓ .

the decisional BDH assumption. Finally, improved efficiency is also possible for BTE schemes achieving chosen-ciphertext security. See [7] for further details.

For completeness, the asymptotic efficiencies of the two constructions secure in the sense of SN-CPA (i.e., in the standard model and in the random oracle model) are given in Table 1.

4 Applications of BTE

We briefly summarize the known applications of BTE.

4.1 Forward Secure Encryption

Cryptographic computations are often carried out on insecure devices for which the threat of key exposure represents a serious and realistic concern. In an effort to mitigate the damage caused by exposure of secret keys stored on such devices, the paradigm of *forward security* was introduced [1, 4]. In a forward-secure scheme, the secret key is updated at regular periods of time (say, at the end of every day), while the public key remains fixed; such schemes guarantee that exposure of the secret key corresponding to a given time period does not enable an adversary to “break” the scheme (in the appropriate sense) for any *prior* time period.

Although a number of forward-secure signature and identification schemes (beginning with [1, 4]) have been proposed, designing a forward-secure (public-key) *encryption* scheme seemed elusive. As outlined in [7], however, BTE schemes can be used to solve this problem, and to give efficient constructions of forward-secure encryption schemes. The basic idea is as follows: let N be the total number of time periods³ for which the system will operate. Each of these time periods is associated with a node in a binary tree of depth $\lceil \log N \rceil$ in the following way: the i^{th} time period will be associated with the i^{th} node of the tree according to a pre-order traversal. We denote this node by $\langle i \rangle$.

The secret key at period i will consist of: (1) the secret key for node $\langle i \rangle$ in the underlying BTE scheme; and also (2) the secret keys (in the underlying BTE

³ We assume for simplicity that N is fixed in advance; in fact, an *unbounded* number of time periods can be supported [7].

scheme) for all right-children of the path from the root of the tree to $\langle i \rangle$. To encrypt a message during period i , a sender simply encrypts it for the node $\langle i \rangle$ (again, using the underlying BTE scheme); note that decryption is possible since the secret key at period i includes, in particular, the secret key for node $\langle i \rangle$. It should be noted also that key updates can be done by erasing the secret key for node $\langle i \rangle$ and using the additional secret keys (i.e., those keys belonging to right-children of the path from the root to $\langle i \rangle$) to derive the necessary keys for the next time period. Details are given in [7], where the following is also proven:

Theorem 3. *(Informal:) Given any BTE scheme secure in the sense of SN-CPA, the above construction yields a forward-secure encryption scheme.*

Since the above construction requires a binary tree of depth $\log N$ to support N time periods, the scheme itself has all parameters at most poly-logarithmic in the number of time periods (cf. Table 1). In fact, additional improvements are possible. These improvements and various extensions of the above theorem, omitted here for lack of space, are given in [7].

4.2 (Hierarchical) Identity-Based Encryption

It was noted earlier that any HIBE scheme is also trivially a BTE scheme, without any modification. Interestingly, one can also show that a BTE scheme is powerful enough to construct a full-fledged HIBE scheme [7] (or, as a special case, an identity-based scheme), albeit under a slightly weaker definition which requires a non-adaptive choice of the target identity. We sketch the construction here. An HIBE of depth t is constructed using a BTE of depth $k \cdot t$, where k is the security parameter. Identities are hashed to strings of length at most $k \cdot t$ by applying a collision-resistant hash function H to the identities at each level; thus, the identity $(id_1 || \dots || id_t)$ is mapped to the string $H(id_1) || \dots || H(id_t)$. It is not hard to show that this gives a secure HIBE, under the relaxed definition of security given above. In fact, because the target identity must be chosen in advance, it is enough for H to be a universal one-way hash function (whose existence is implied by any BTE scheme); thus, we have:

Theorem 4. *Assuming the existence of a BTE scheme secure in the sense of SN-CPA, there exists an HIBE scheme of arbitrary polynomial depth secure under a non-adaptive choice of target identity.*

4.3 Chosen-Ciphertext Security

Recently, an interesting connection between identity-based encryption and security against chosen-ciphertext attacks (for “standard” public-key encryption schemes) has been shown [8]. In particular, it was shown how any identity-based encryption scheme can be used to give a simple and efficient construction of a regular encryption scheme secure against chosen-ciphertext attacks (i.e., a “CCA-secure” encryption scheme). The resulting construction avoids the use of

any generic NIZK proofs, and furthermore seems not to follow the paradigm of all previously-known constructions of CCA-secure encryption schemes (cf. [12]).

We describe the construction of a CCA-secure standard encryption scheme now: The public key of the scheme will be the master public key PK of the identity-based scheme, while the secret key SK is the corresponding master secret key. To encrypt a message m , the sender generates verification/signature keys (vk, sk) for any one-time signature scheme, and encrypts m for “identity” vk (using the underlying identity-based scheme). The sender signs the resulting ciphertext C to obtain a signature σ , and sends $\langle vk, C, \sigma \rangle$ to the receiver.

The receiver first verifies that σ is a correct signature on C with respect to vk ; if not, the ciphertext is rejected. Otherwise, the receiver uses the master secret key SK to generate a decryption key SK_{vk} corresponding to the “identity” vk . Notice that it can then decrypt C using SK_{vk} .

The following is proven in [8]:

Theorem 5. *(Informal:) Given any identity-based encryption scheme secure under a non-adaptive choice of target identity, the above construction yields a CCA-secure encryption scheme.*

Note that the identity-based scheme need only be secure against a *non-adaptive* choice of target identity; combined with Theorems 1 and 4, this results in a new construction of a CCA-secure encryption scheme in the standard model.

It has already been noted above that a similar technique may be used to construct a BTE scheme secure against chosen-ciphertext attacks, starting with any BTE scheme secure in the sense of SN-CPA. Because this gives a slightly stronger result than that of Theorem 2, we state it here for completeness:

Theorem 6. *Assuming the existence of a BTE scheme secure in the sense of SN-CPA, there exists a BTE scheme secure in the sense of SN-CCA.*

See [8] for further details.

4.4 Adaptively-Secure Encryption

Standard definitions of secure encryption do not guarantee security in the case of *adaptive* corruptions. In a setting where such adaptive corruptions are possible, the encryption scheme should provide the additional guarantee that the information gathered by an adversary when corrupting parties (and, in particular, learning their secret keys) does not give it any additional advantage toward compromising the security of the uncorrupted parties. Very roughly speaking, this requirement may be captured by the existence of a simulator that can generate “dummy” ciphertexts which it can later “open” (by revealing an appropriate secret key) to any given message. (Of course, this must be done in such a way that the revealed secret key “matches” the fixed public key.) Schemes achieving this notion of security are termed *adaptively secure*. We do not further motivate or describe the definition, but instead refer the reader elsewhere [2, 6, 21, 15, 9] for additional discussion.

Nielsen has shown [21] that non-interactive adaptively-secure encryption (in the standard model) is “impossible” unless the secret key is as long as the length of all plaintext messages that are sent. In particular, this implies that no adaptively-secure scheme supporting an a priori *unbounded* number of messages is possible. This result is in stark contrast to the case for encryption schemes which are not adaptively secure.

It is possible to circumvent Nielsen’s impossibility result, however, by considering *key-evolving* cryptosystems; i.e., those in which the secret key evolves over time. This suggests using forward-secure encryption as a building block toward building adaptively-secure schemes. (Note that a forward-secure encryption scheme, by itself, is not necessarily adaptively secure.) In fact, a construction of an adaptively-secure encryption scheme based on any forward-secure encryption scheme has recently been given [9]:

Theorem 7. (*Informal:*) *Assuming the existence of a forward-secure encryption scheme and an NIZK proof system for NP, there exists a non-interactive adaptively-secure encryption scheme for an unbounded number of messages.*

Since both a forward-secure encryption scheme as well as NIZK proofs may be based on the BDH assumption (cf. [7]), the BDH assumption suffices to imply the result of the theorem.

More efficient constructions of adaptively-secure encryption, which avoid NIZK proofs altogether (but which in some cases require additional number theoretic assumptions), are also given in [9].

Acknowledgments

It was great fun to collaborate with Ran Canetti and Shai Halevi on the all work described herein. I would also like to thank the program chairs for ICISC 2003 (Jong In Lim and Dong Hoon Lee) for inviting me to present this survey, and for a very enjoyable visit to Korea.

References

1. R. Anderson. Two Remarks on Public Key Cryptology. Invited Lecture, *4th ACM Conference on Computer and Communications Security*, 1997. Available at <http://www.cl.cam.ac.uk/ftp/users/rja14/forwardsecure.pdf>.
2. D. Beaver and S. Haber. Cryptographic Protocols Secure Against Dynamic Adversaries. *Adv. in Cryptology — Eurocrypt 1992*, LNCS vol. 658, Springer-Verlag, pp. 307–323, 1992.
3. M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway. Relations Among Notions of Security for Public-Key Encryption Schemes. *Adv. in Cryptology — Crypto 1998*, LNCS vol. 1462, Springer-Verlag, pp. 26–45, 1998.
4. M. Bellare and S. Miner. A Forward-Secure Digital Signature Scheme. *Adv. in Cryptology — Crypto 1997*, LNCS vol. 1666, Springer-Verlag, pp. 75–89, 1997.

5. D. Boneh and M. Franklin. Identity-Based Encryption from the Weil Pairing. *Adv. in Cryptology — Crypto 2001*, LNCS vol. 2139, Springer-Verlag, pp. 213–229, 2001. Full version to appear in *SIAM J. Computing* and available at <http://eprint.iacr.org/2001/090>.
6. R. Canetti, U. Feige, O. Goldreich, and M. Naor. Adaptively Secure Multi-Party Computation. *28th ACM Symposium on Theory of Computing (STOC)*, ACM, pp. 639–648, 1996.
7. R. Canetti, S. Halevi, and J. Katz. A Forward-Secure Public-Key Encryption Scheme. *Adv. in Cryptology — Eurocrypt 2003*, LNCS vol. 2656, Springer-Verlag, pp. 255–271, 2003. Full version available at <http://eprint.iacr.org/2003/083>.
8. R. Canetti, S. Halevi, and J. Katz. Chosen-Ciphertext Security from Identity-Based Encryption. Available at <http://eprint.iacr.org/2003/182>.
9. R. Canetti, S. Halevi, and J. Katz. Flash Encryption: Adaptive and Forward Security with Short Keys. Manuscript, November 2003.
10. C. Cocks. An Identity-Based Encryption Scheme Based on Quadratic Residues. *Cryptography and Coding*, LNCS vol. 2260, Springer-Verlag, pp. 360–363, 2001.
11. Y. Dodis, J. Katz, S. Xu, and M. Yung. Strong Key-Insulated Signature Schemes. *PKC 2003*, LNCS vol. 2567, pp. 130–144, Springer-Verlag, 2003.
12. E. Elkin and A. Sahai. A Unified Methodology For Constructing Public-Key Encryption Schemes Secure Against Adaptive Chosen-Ciphertext Attack. To appear, *1st Theory of Cryptography Conference*, 2004. Available at <http://eprint.iacr.org/2002/042>.
13. A. Fiat and A. Shamir. How to Prove Yourself: Practical Solutions to Identification and Signature Problems. *Adv. in Cryptology — Crypto 1986*, LNCS vol. 263, Springer-Verlag, pp. 186–194, 1986.
14. C. Gentry and A. Silverberg. Hierarchical Identity-Based Cryptography. *Adv. in Cryptology — Asiacrypt 2002*, LNCS vol. 2501, Springer-Verlag, pp. 548–566, 2002.
15. O. Goldreich. Draft of a Chapter on Cryptographic Protocols. Manuscript, June 2003. Available at <http://www.wisdom.weizmann.ac.il/~oded/foc-vol2.html>
16. L. Guillou and J.-J. Quisquater. A “Paradoxical” Identity-Based Signature Schemes Resulting from Zero-Knowledge. *Adv. in Cryptology — Crypto 1988*, LNCS vol. 403, Springer-Verlag, pp. 216–231, 1988.
17. J. Horwitz and B. Lynn. Toward Hierarchical Identity-Based Encryption. *Adv. in Cryptology — Eurocrypt 2002*, LNCS vol. 2332, Springer-Verlag, pp. 466–481, 2002.
18. A. Joux. A One-Round Protocol for Tri-Partite Diffie Hellman. *Fourth Algorithmic Number Theory Symposium (ANTS)*, LNCS vol. 1838, Springer-Verlag, pp. 385–394, 2000.
19. A. Joux and K. Nguyen. Separating Decision Diffie-Hellman from Diffie-Hellman in Cryptographic Groups. Manuscript, January 2001. Available at <http://eprint.iacr.org/2001/003/>.
20. M. Naor and M. Yung. Public Key Cryptosystems Provably Secure Against Chosen-Ciphertext Attacks. *22nd ACM Symposium on Theory of Computing (STOC)*, ACM, pp. 427–437, 1990.
21. J.B. Nielsen. Separating Random Oracle Proofs from Complexity-Theoretic Proofs: The Non-Committing Encryption Case. *Adv. in Cryptology — Crypto 2002*, LNCS vol. 2442, Springer-Verlag, pp. 111–126, 2002.
22. A. Sahai. Non-malleable Non-Interactive Zero-Knowledge and Adaptive Chosen-Ciphertext Security. *40th IEEE Symposium on Foundations of Computer Science (FOCS)*, IEEE, pp. 543–553, 1999.
23. A. Shamir. Identity-Based Cryptosystems and Signature Schemes. *Adv. in Cryptology — Crypto 1984*, LNCS vol. 196, Springer-Verlag, pp. 47–53, 1984.