Function-Private Identity-Based Encryption: Hiding the Function in Functional Encryption*

Dan Boneh, Ananth Raghunathan, and Gil Segev

Computer Science Department
Stanford University, Stanford, CA 94305

Abstract. We put forward a new notion, *function privacy*, in identity-based encryption and, more generally, in functional encryption. Intuitively, our notion asks that decryption keys reveal essentially no information on their corresponding identities, beyond the absolute minimum necessary. This is motivated by the need for providing *predicate privacy* in public-key searchable encryption. Formalizing such a notion, however, is not straightforward as given a decryption key it is always possible to learn some information on its corresponding identity by testing whether it correctly decrypts ciphertexts that are encrypted for specific identities.

In light of such an inherent difficulty, any meaningful notion of function privacy must be based on the *minimal* assumption that, from the adversary's point of view, identities that correspond to its given decryption keys are sampled from somewhat unpredictable distributions. We show that this assumption is in fact *sufficient* for obtaining a strong and realistic notion of function privacy. Loosely speaking, our framework requires that a decryption key corresponding to an identity sampled from any sufficiently unpredictable distribution is indistinguishable from a decryption key corresponding to an independently and uniformly sampled identity.

Within our framework we develop an approach for designing function-private identity-based encryption schemes, leading to constructions that are based on standard assumptions in bilinear groups (DBDH, DLIN) and lattices (LWE). In addition to function privacy, our schemes are also anonymous, and thus yield the first public-key searchable encryption schemes that are provably *keyword private*: A search key sk_w enables to identify encryptions of an underlying keyword w, while not revealing any additional information about w beyond the minimum necessary, as long as the keyword w is sufficiently unpredictable.

1 Introduction

Public-key searchable encryption is needed when a proxy is asked to route encrypted messages based on their content. For example, consider a payment gateway that needs to route transactions based on the transaction type. Transactions for benign items are routed for quick processing while transactions for sensitive

* Due to space limitations the reader is referred to the full version [19].

R. Canetti and J.A. Garay (Eds.): CRYPTO 2013, Part II, LNCS 8043, pp. 461–478, 2013.

items are routed for special processing. Similarly, consider an email gateway that routes emails based on the contents of the subject line. Urgent emails are routed to the user's mobile device, while less urgent mails are routed to the user's desktop. When the data is encrypted a simple design is to give such gateways full power to decrypt all ciphertexts, but this clearly exposes more information than necessary.

A better solution, called public-key searchable encryption (introduced by Boneh, Di Crescenzo, Ostrovsky and Persiano [17]), is to give the gateway a trapdoor that enables it to learn the information it needs and nothing else. In recent years many elegant public-key searchable encryption systems have been developed [17,36,1,21,47,39,6,24,2,4] supporting a wide variety of search predicates.

Private Searching. Beyond the standard notions of data privacy, it is often also necessary to guarantee *predicate privacy*, i.e., to keep the specific search predicate hidden from the gateway. For example, in the payment scenario it may be desirable to keep the list of sensitive items secret, and in the email scenario users may not want to reveal the exact criteria they use to classify an email as urgent. Consequently, we want the trapdoor given to the gateway to reveal as little as possible about the search predicate.

While this question has been considered before [48,44,14,46], it is often noted that such a notion of privacy cannot be achieved in the public-key setting. For example, to test if an email from "spouse" is considered urgent the gateway could simply use the public key to create an email from the spouse and test if the trapdoor classifies it as urgent. More generally, the gateway can encrypt messages of its choice and apply the trapdoor to the resulting ciphertexts, thereby learning how the search functionality behaves on these messages. Hence, leaking some information about the search predicate is unavoidable.

As a concrete example, consider the case of keyword search [17]: A search key sk_w corresponds to a particular keyword w, and the search matches a ciphertext $\mathsf{Enc}(pk, m)$ if and only if $m = w$. In this case, it may be possible to formalize and realize a notion of "private keyword search" asking that a search key reveals no more information than what can be learned by invoking the search algorithm.

Function-private IBE: A New Notion of Security. Motivated by the challenge of hiding the search predicates in public-key searchable encryption, in this paper we introduce a new notion of security, *function privacy*, for identity-based encryption.[1] The standard notion of security for anonymous IBE schemes (e.g., [18,22,31,32,3,11]), asks that a ciphertext $c = \mathsf{Enc}(\mathrm{pp}, \mathrm{id}, m)$ reveals essentially

[1] As observed by Abdalla et al. [1], any anonymous IBE scheme can be used as a public-key searchable encryption scheme by defining the search key sk_w for a keyword w as the IBE secret key for the identity $\mathrm{id} = w$. A keyword w' is encoded as $c = \mathsf{Enc}(\mathrm{pp}, w', 0)$ and one tests if c matches the keyword w by invoking the IBE decryption algorithm on c with the secret key sk_w. The IBE anonymity property ensures that c reveals nothing else about the payload w'. For this reason we focus on *anonymous* IBE schemes, although we note that our notion of function privacy does not require anonymity.

no information on the pair (id, m) as long as a secret key $\mathrm{sk}_{\mathrm{id}}$ corresponding to the identity id is not explicitly provided (but secret keys corresponding to other identities may be provided). Our notion of function privacy takes a step forward by asking that it should not be possible to learn any information, beyond the absolute minimum necessary, on the identity id corresponding to a given secret key $\mathrm{sk}_{\mathrm{id}}$.

Formalizing a realistic notion of function privacy, however, is not straightforward due to the actual functionality of identity-based encryption. Specifically, assuming that an adversary who is given a secret key $\mathrm{sk}_{\mathrm{id}}$ has some *a priori* information that the corresponding identity id belongs to a small set S of identities (e.g., $S = \{\mathrm{id}_0, \mathrm{id}_1\}$), then the adversary can fully recover id: The adversary simply needs to encrypt a (possibly random) message m for each $\mathrm{id}' \in S$, and then run the decryption algorithm on the given secret key $\mathrm{sk}_{\mathrm{id}}$ and each of the resulting ciphertexts $c' = \mathsf{Enc}(\mathrm{pp}, \mathrm{id}', m)$ to identify the one that decrypts correctly. In fact, as long as the adversary has some *a-priori* information according to which the identity id is sampled from a distribution whose min-entropy is at most logarithmic in the security parameter, there is a non-negligible probability for a full recovery.

Our Contributions. In light of the above inherent difficulty, any notion of function privacy for IBE schemes would have to be based on the *minimal* assumption that, from the adversary's point of view, identities that correspond to its given secret keys are sampled from distributions with a certain amount of min-entropy (which has to be at least super-logarithmic in the security parameter). Our work shows that this necessary assumption is in fact *sufficient* for obtaining a strong and meaningful indistinguishability-based notion of function privacy.

Our work formalizes this new notion of security (we call it *function privacy* to emphasize the fact that $\mathrm{sk}_{\mathrm{id}}$ hides the functionality that it provides). Loosely speaking, our basic notion of function privacy requires that a secret key $\mathrm{sk}_{\mathrm{id}}$, where id is sampled from any sufficiently unpredictable (adversarially-chosen) distribution,[2] is indistinguishable from a secret key corresponding to an independently and uniformly sampled identity. In addition, we also consider a stronger notion of function privacy, to which we refer as *enhanced* function privacy. This enhanced notion addresses the fact that in various applications (such as searching on encrypted data), an adversary may obtain not only a secret key $\mathrm{sk}_{\mathrm{id}}$, but also encryptions $\mathsf{Enc}(\mathrm{pp}, \mathrm{id}, m)$ of messages m. Our notion of enhanced function privacy asks that even in such a scenario, it should not be possible to learn any unnecessary information on the identity id.

[2] We emphasize that the distribution is allowed to depend on the public parameters of the scheme. This is in contrast to the setting of deterministic public-key encryption (DPKE) [8], where similar inherent difficulties arise when formalizing notions of security. Nevertheless, our notion is inspired by that of [8], and we refer the reader to Section 2 for an elaborate discussion (in particular, we discuss a somewhat natural DPKE-based approach for designing function-private IBE schemes which fails to satisfy our notion of security and only satisfies a weaker, less realistic, one).

We refer the reader to Section 2 for the formal definitions, and for descriptions of simple attacks exemplifying that the anonymous IBE schemes presented in [18,32,3,40] do not satisfy even our basic notion of function privacy.[3]

Within our framework we develop an approach for designing identity-based encryption schemes that satisfy our notions of function private. Our approach leads to constructions that are based on standard assumptions in bilinear groups (DBDH, DLIN) and lattices (LWE). In particular, our schemes yield keyword searchable public-key encryption schemes that *do not reveal the keywords*: A search key sk_w reveals nothing about its corresponding keyword w beyond the minimum necessary, as long as the keyword w is chosen from a sufficiently unpredictable distribution.

The Bigger Picture: Functional Encryption and Obfuscation. Our notion of function privacy for IBE naturally generalizes to functional encryption systems [20,43,12,37,5,34], where we obtain an additional security requirement on such systems. Here, a functional secret key sk_f corresponding to a function f enables to compute $f(m)$ given an encryption $c = \mathsf{Enc}_{pk}(m)$. Functional encryption systems, however, need not be predicate private and sk_f may leak unnecessary information about f. Intuitively, we say that a functional encryption system is *function private* if such a functional secret key sk_f does not reveal information about f beyond what is already known and what can be obtained by running the decryption algorithm on test ciphertexts. This can be formalized within a suitable framework of program obfuscation (e.g., [25,7,41,35,50,26] and the references therein) by asking, for example, that any adversary that receives a functional secret key sk_f learns no more information than a simulator that has oracle access to the function f.

In this setting, our identity-based encryption schemes provide function privacy for the class of functions defined as

$$f_{\mathrm{id}^*}(\mathrm{id}, m) = \begin{cases} m \text{ if } \mathrm{id} = \mathrm{id}^* \\ \perp \text{ otherwise} \end{cases}$$

where id^* is sampled from an unpredictable distribution. A fascinating direction for future work is to extend our results to more general classes of functions.

Non-Adaptive Function Privacy and Deterministic Encryption. The inherent difficulty discussed above in formalizing function privacy is somewhat similar to the one that arises in the context of deterministic public-key encryption (DPKE), introduced by Bellare, Boldyreva, and O'Neill [8] (see also [10,15,9,23,30,42,51,45]). In that setting one would like to capture as-strong-as-possible notions of security that can be satisfied by public-key encryption

[3] We note that other anonymous IBE schemes, such as [31,22,11] for which we were not able to find such simple attacks, can always be *assumed* to be function private based on somewhat non-standard entropy-based assumptions (such assumptions would essentially state that the schemes satisfy our definition). In this paper we are interested in schemes whose function privacy can be based on standard assumptions (e.g., DBDH, DLIN, LWE).

schemes whose encryption algorithms are deterministic. Similarly to our setting, if an adversary has some *a priori* information that a ciphertext $c = \mathsf{Enc}_{pk}(m)$ corresponds to a plaintext m that is sampled from a low-entropy source (e.g., $m \in \{m_0, m_1\}$), then the plaintext can be fully recovered: the adversary simply needs to encrypt all "likely" plaintexts and to compare each of the resulting ciphertexts to c. Therefore, any notion of security for DPKE has to be based on the assumption that plaintexts are sampled from distributions with a certain amount of min-entropy (which has to be at least super-logarithmic in the security parameter).

However, unlike in our setting, in the setting of DPKE it is also necessary to limit the dependency of plaintexts on the public-key of the scheme.[4] In our setting, as the key-generation algorithm is allowed to be randomized, such limitations are not inherent: we allow adversaries to specify identity distributions in an adaptive manner after seeing the public parameters of the scheme.

This crucial difference between our setting and the setting of DPKE rules out, in particular, the following natural approach for designing anonymous IBE schemes providing function privacy: encapsulate all identities with a DPKE scheme, and then use any existing anonymous IBE scheme treating the ciphertexts of the DPKE scheme as its identities. That is, for encrypting to identity id, first encrypt id using a DPKE scheme and then treat the resulting ciphertext as an identity for an anonymous IBE system. This approach clearly preserves the standard security of the underlying IBE scheme. Moreover, as secret keys are now generated as sk_c, where $c = \mathsf{Enc}_{pk}(\mathrm{id})$ is a deterministic encryption of id, instead of as $\mathrm{sk}_{\mathrm{id}}$, one could hope that $\mathrm{sk}_{\mathrm{id}}$ does not reveal any unnecessary information on id as long as id is sufficiently unpredictable.

This approach, however, fails to satisfy our notion of function privacy and only satisfies a weaker,"non-adaptive", one.[5] Specifically, the notion of function privacy that is satisfied by such a two-tier construction is that secret keys do not reveal any unnecessary information on their corresponding identities as long as the identities are essentially independent of the public parameters of the scheme. In the full version [19] we formalize this non-adaptive notion and present a generic transformation satisfying it based on any IBE scheme. In fact, observing that the DPKE-based construction described above never actually uses the decryption algorithm of the DPKE scheme, in our generic transformation we show that above idea can be realized without using a DPKE scheme. Instead, we only need to assume the existence of collision-resistant hash functions (and also use any pairwise independent family of permutations).

[4] Intuitively, the reason is that plaintexts distributions that can depend on the public key can use any deterministic encryption algorithm as a subliminal channel for leaking information on the plaintexts (consider, for example, sampling a uniform plaintext m for which the most significant bit of $c = \mathsf{Enc}_{pk}(m)$ agrees with that of m). We refer the reader to [8,45] for an in-depth discussion.

[5] As discussed above, any DPKE becomes insecure once plaintext distributions (which here correspond to identity distributions) are allowed to depend on the public key of the scheme.

1.1 Our Approach: "Extract-Augment-Combine"

Our approach consists of three main steps: "extract," "augment," and "combine." We begin with a description of the main ideas underlying each step, and then provide an example using a concrete IBE scheme.

Given any anonymous IBE scheme $\Pi = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$, we use the exact same setup algorithm Setup, and our first step is to modify its key-generation algorithm KeyGen as follows: Instead of generating a secret key for an identity id, first apply a strong randomness extractor Ext to id using a randomly chosen seed s, then generate a secret key $\mathrm{sk}_{\mathrm{id}_s}$ for the identity $\mathrm{id}_s := \mathsf{Ext}(\mathrm{id}, s)$, and output the pair $(s, \mathrm{sk}_{\mathrm{id}_s})$ as a secret for id in the new scheme. This steps clearly guarantees function privacy: as long as the identity id is sampled from a sufficiently unpredictable distribution,[6] the distribution (s, id_s) is statistically close to uniform, and therefore the pair $(s, \mathrm{sk}_{\mathrm{id}_s})$ reveals no information on the identity id.

This extraction step, however, may hurt the data privacy of the underlying scheme. For example, since randomness extractors are highly non-injective by definition, an adversary that is given a secret key $(s, \mathrm{sk}_{\mathrm{id}_s})$ may be able to find an identity id' such that $\mathsf{Ext}(\mathrm{id}, s) = \mathsf{Ext}(\mathrm{id}', s)$. In this case, the same secret key is valid for both id and id', contradicting the data privacy of the resulting scheme. Therefore, for overcoming this problem we make sure that the extractor is *at least* collision resistant: although many collisions exist, a computationally-bounded adversary will not be able to find one. This is somewhat natural to achieve in the random-oracle model [13], but significantly more challenging in the standard model.

An even more challenging problem is that the extraction step hurts the decryption of the underlying scheme. Specifically, when encrypting a message m for an identity id, the encryption algorithm does not know which seed s will be chosen (or was already chosen) when generating a secret key for id. In other words, the correctness of the decryption algorithm Dec should hold for any choice of seed s by the key-generation algorithm KeyGen, although s is not known to the encryption algorithm Enc. One possibility, is to modify the encryption algorithm such that it outputs an encryption of m for id_s for all possible seeds s. This clearly fails, as the number of seeds is inherently super-polynomial in the security parameter. We overcome this problem by augmenting ciphertexts of the underlying scheme with various additional pieces of information. These will enable the new decryption algorithm to combine the pieces in a particular way for generating an encryption of m for the identity id_s for any given s, and then simply apply the underlying decryption algorithm using the specific seed s chosen by the key-generation algorithm.[7]

[6] Note that the new scheme assumes a slightly larger identity space compared to the underlying scheme.

[7] In fact, in some of our schemes the decryption algorithm combines the pieces to generate an encryption of a related message m' from which m can be easily recovered (e.g., $m' = 2m$).

Our approach introduces the following two main challenges that we overcome in each of our constructions:

- Augmenting the ciphertexts of the underlying scheme with additional pieces of information may hurt the data privacy of the underlying scheme.
- Combining the additional pieces of information for generating an encryption for id_s for any given s requires using an extractor Ext that exhibits a particular interplay with the underlying encryption and decryption algorithms.

Our constructions in this paper are obtained by applying our approach to various known anonymous IBE schemes [18,32,3,40]. To do so, we overcome the two main challenges mentioned above in ways that are "tailored" specifically to each scheme. Using our approach we provide the following constructions (see also Table 1):

- In the random-oracle model we give fully-secure constructions from pairings and lattices by building upon the systems of Boneh and Franklin [18] (based on the DBDH assumption) and of Gentry, Peikert and Vaikuntanathan [32] (based on the LWE assumption).
- In the standard model we give selectively-secure constructions from pairings and lattices based on the constructions of Agrawal, Boneh and Boyen [3] (based on the LWE assumption) and of Kurosawa and Phong [40] (based on the DLIN assumption), which we then generalize to a fully-secure construction (based on the DLIN assumption[8]).

In all instances our constructions are based on the same complexity assumptions as the underlying systems.

Table 1. Our IBE schemes

Scheme	Model	Data Privacy	Function Privacy
DBDH	Random Oracle	Full	Statistical
LWE1	Random Oracle	Full	Statistical
DLIN1	Standard	Selective	Statistical + Non-adaptive enhanced
LWE2	Standard	Selective	Statistical
DLIN2	Standard	Full	Statistical + Enhanced
CRH	Standard	Full	Non-adaptive statistical enhanced

A Concrete Example. We conclude this section by exemplifying our approach using our DBDH-based construction in the random-oracle model. (We refer the reader to the full version [19] for a more formal description of the scheme and its proofs of data privacy and function privacy.) The scheme is obtained by applying our approach to the anonymous IBE scheme of Boneh and Franklin [18].

[8] We note that a similar generalization can also be applied to our selectively-secure LWE-based scheme in the standard model.

- The setup algorithm in the scheme of Boneh and Franklin samples $\alpha \leftarrow \mathbb{Z}_p^*$, and lets $h = g^\alpha$, where g is a generator of a group $\mathbb{G}$ of prime order p. The public parameters are g and h, and the master secret key is α. Our scheme has exactly the same setup algorithm.
- The key-generation algorithm in the scheme of Boneh and Franklin computes a secret key for an identity id as $\mathrm{sk}_{\mathrm{id}} = H(\mathrm{id})^\alpha$, where H is a random oracle mapping identities into the group $\mathbb{G}$. As discussed above our first step is to extract from id. First, we use a random oracle mapping identities into $\mathbb{G}^\ell$ for some $\ell > 1$. Then, for $H(\mathrm{id}) = (h_1, \ldots, h_\ell) \in \mathbb{G}^\ell$, we sample an extractor seed $s = (s_1, \ldots, s_\ell) \leftarrow \mathbb{Z}_p^\ell$, and output the secret key $(s, (\mathsf{Ext}(H(\mathrm{id}), s)^\alpha)$ where we use the specific extractor $\mathsf{Ext}((h_1, \ldots, h_\ell), (s_1, \ldots, s_\ell)) = \prod_{j=1}^{\ell} h_j^{s_j}$. Note that Ext is, in particular, collision resistant based on the discrete logarithm assumption in the group $\mathbb{G}$.
- An encryption of a message m for an identity id in the scheme of Boneh and Franklin is a pair (c_0, c_1), defined as $c_0 = g^r$ and $c_1 = \hat{e}(h, H(\mathrm{id}))^r \cdot m$. In our scheme, an encryption of a message m for an identity id consists of $\ell + 1$ components $(c_0, \ldots, c_\ell)$ defined as $c_0 = g^r$, and $c_i = \hat{e}(h, h_i)^r \cdot m$ for every $i \in [\ell]$, where $H(\mathrm{id}) = (h_1, \ldots, h_\ell)$. This is exactly using the encryption algorithm of Boneh and Franklin for separately encrypting m for each of the h_i's while re-using the same randomness r. The main technical challenge that is left is showing that such augmented ciphertexts still provide data privacy.
- Our decryption algorithm on input a ciphertext $c = (c_0, \ldots, c_\ell)$, and a secret key $\mathrm{sk}_{\mathrm{id}} = (s_1, \ldots, s_\ell, z)$, combines $c_1, \ldots, c_\ell$ by computing

$$\prod_{i=1}^{\ell} c_i^{s_i} = \hat{e}(h, \prod_{i=1}^{\ell} h_i^{s_i})^r \cdot m^{s_1 + \cdots + s_\ell} = \hat{e}(h, \mathrm{id}_s)^r \cdot m^{s_1 + \cdots + s_\ell},$$

  where $\mathrm{id}_s = \mathsf{Ext}(H(\mathrm{id}), s)$, as before. Note that the pair $(c_0, \prod_{i=1}^{\ell} c_i^{s_i})$ is exactly an encryption of the message $m' = m^{s_1 + \cdots + s_\ell}$ for the identity id_s in the scheme of Boneh and Franklin. This allows to invoke the decryption algorithm of Boneh and Franklin for recovering m', and then to easily recover m (as the s_i's are given in the clear).

1.2 Related Work

Searchable encryption has been studied in both the symmetric settings [48,29,46] and public-key settings [17,36,1,21,47,39,6,24,4]. Public-key searching on encrypted data now supports equality testing, disjunctions and conjunctions, range queries, CNF/DNF formulas, and polynomial evaluation. These schemes, however, are not function private in that their secret searching keys reveal information about their corresponding predicates. Indeed, until this work, predicate privacy seemed impossible in the public-key settings.

The impossibility argument does not apply in the symmetric key settings where the encryptor and decryptor have a shared secret key. In this setting

the entity searching over ciphertexts does not have the secret key and cannot (passively) test the searching key on ciphertexts of its choice. Indeed, in the symmetric-key setting predicate privacy is possible and a general solution to private searching on encrypted data was provided by Goldreich and Ostrovsky [33] in their construction of an oblivious RAM. More efficient constructions are known for equality testing [48,27,29,28,49,38] and inner product testing [46]. The latter enables CNF/DNF formulas, polynomial evaluation, and exact thresholds.

A closely related problem called *private stream searching* asks for the complementary privacy requirements: the data is available in the clear, but the search predicate must remain hidden. Constructions in these settings support efficient equality testing [44,14] and can be viewed as a more expressive variant of private information retrieval.

1.3 Notation

Throughout the paper we use the following standard notation. For an integer $n \in \mathbb{N}$ we denote by $[n]$ the set $\{1, \ldots, n\}$, and by U_n the uniform distribution over the set $\{0,1\}^n$. For a random variable X we denote by $x \leftarrow X$ the process of sampling a value x according to the distribution of X. Similarly, for a finite set S we denote by $x \leftarrow S$ the process of sampling a value x according to the uniform distribution over S. We denote by $\mathbf{x}$ (and sometimes $\boldsymbol{x}$) a vector $(x_1, \ldots, x_{|\mathbf{x}|})$. We denote by $\mathbf{X} = (X_1, \ldots, X_T)$ a joint distribution of T random variables, and by $\mathbf{x} = (x_1, \ldots, x_T)$ a sample drawn from $\mathbf{X}$. For two bit-strings x and y we denote by $x \| y$ their concatenation. A non-negative function $f : \mathbb{N} \to \mathbb{R}$ is negligible if it vanishes faster than any inverse polynomial. For a real number $x \in \mathbb{R}$ we define $\lfloor x \rceil = \lfloor x + 1/2 \rfloor$ (i.e., the nearest integer to x). For a group $\mathbb{G}$ of order p with generator g and any $\mathbf{X} \in \mathbb{Z}_p^{n \times m}$, we denote the matrix whose (i,j)-th entry is $(g^{x_{i,j}})$ by $g^{\mathbf{X}}$.

The *min-entropy* of a random variable X is $\mathbf{H}_\infty(X) = -\log(\max_x \Pr[X = x])$. A *k-source* is a random variable X with $\mathbf{H}_\infty(X) \geq k$. A $(k_1, \ldots, k_T)$*-source* is a random variable $\mathbf{X} = (X_1, \ldots, X_T)$ where each X_i is a k_i-source. A (T, k)*-block-source* is a random variable $\mathbf{X} = (X_1, \ldots, X_T)$ where for every $i \in [T]$ and $x_1, \ldots, x_{i-1}$ it holds that $X_i|_{X_1 = x_1, \ldots, X_{i-1} = x_{i-1}}$ is a k-source.

1.4 Paper Organization

The remainder of this paper is organized as follows. In Section 2 we formally define our notion of function privacy for identity-based encryption. In Section 3 we present a selectively-secure DLIN-based scheme in the standard model, and in Section 4 we discuss several extensions and open problems. Due to space limitations we refer the reader to the full version [19].

2 Modeling Function Privacy for IBE

In this section we introduce our notions of function privacy for anonymous IBE schemes.[9] Recall that the standard notion of security for anonymous IBE schemes, anon-IND-ID-CPA, asks that a ciphertext $c = \mathsf{Enc}(\mathrm{pp}, \mathrm{id}, m)$ reveals essentially no information on the pair (id, m) as long as a secret key $\mathrm{sk}_{\mathrm{id}}$ corresponding to the identity id is not explicitly provided (but secret keys corresponding to other identities may be provided). We refer to this notion of security as *data privacy*. As discussed in Section 1, we put forward three notions of function privacy: a basic notion, an "enhanced" notion, and a non-adaptive notion. Due to space limitations, in this section we focus on our basic notion, and refer the reader to the full version [19] for our enhanced and non-adaptive notions.

Throughout this section we let T, k, and $k_1, \ldots, k_T$ be functions of the security parameter $\lambda \in \mathbb{N}$. In addition, we note that in the random-oracle model, all algorithms, adversaries, oracles, and distributions are given access to the random oracle.

Our basic notion of function privacy asks that it should not be possible to learn any information, beyond the absolute minimum necessary, on the identity id corresponding to a given secret key $\mathrm{sk}_{\mathrm{id}}$. Specifically, our notion considers adversaries that are given the public parameters of the scheme, and can interact with a "real-or-random" function-privacy oracle $\mathsf{RoR}^{\mathsf{FP}}$. This oracle takes as input any adversarially-chosen distribution over vectors of identities, and outputs secret keys either for identities sampled from the given distribution or for independently and uniformly distributed identities.[10] We allow adversaries to adaptively interact with the real-or-random oracle, for any polynomial number of queries, as long as the distributions have a certain amount of min-entropy. At the end of the interaction, we ask that adversaries have only a negligible probability of distinguishing between the "real" and "random" modes of the oracle. The following definitions formally capture our basic notion of function privacy.

Definition 2.1 (Real-or-random function-privacy oracle). *The real-or-random function-privacy oracle* $\mathsf{RoR}^{\mathsf{FP}}$ *takes as input triplets of the form* $(\mathsf{mode}, \mathrm{msk}, \boldsymbol{ID})$*, where* $\mathsf{mode} \in \{\mathsf{real}, \mathsf{rand}\}$*,* msk *is a master secret key, and* $\boldsymbol{ID} = (ID_1, \ldots, ID_T) \in \mathcal{ID}^T$ *is a circuit representing a joint distribution over* $\mathcal{ID}^T$*. If* $\mathsf{mode} = \mathsf{real}$ *then the oracle samples* $(\mathrm{id}_1, \ldots, \mathrm{id}_T) \leftarrow \boldsymbol{ID}$ *and if* $\mathsf{mode} = \mathsf{rand}$ *then the oracle samples* $(\mathrm{id}_1, \ldots, \mathrm{id}_T) \leftarrow \mathcal{ID}^T$ *uniformly. It then invokes the algorithm* $\mathsf{KeyGen}(\mathrm{msk}, \cdot)$ *on each of* $\mathrm{id}_1, \ldots, \mathrm{id}_T$ *and outputs a vector of secret keys* $(\mathrm{sk}_{\mathrm{id}_1}, \ldots, \mathrm{sk}_{\mathrm{id}_T})$*.*

Definition 2.2 (Function-privacy adversary). *Let* $X \in \{(T, k)\text{-}block, (k_1, \ldots, k_T)\}$*. An* X*-source function-privacy adversary* $\mathcal{A}$ *is an algorithm that is*

[9] We focus on *anonymous* IBE schemes as our motivating application is public-key *searchable* encryption, to which anonymity is crucial [1].

[10] We note that the resulting notion of security is polynomially equivalent to the one obtained by using a "left-or-right" oracle instead of a "real-or-random" oracle, as for example, in the case of semantic security for public-key encryption schemes.

given as input a pair $(1^\lambda, \mathrm{pp})$ *and oracle access to* $\mathsf{RoR}^{\mathsf{FP}}(\mathsf{mode}, \mathrm{msk}, \cdot)$ *for some* $\mathsf{mode} \in \{\mathsf{real}, \mathsf{rand}\}$*, and to* $\mathsf{KeyGen}(\mathrm{msk}, \cdot)$*, and each of its queries to* $\mathsf{RoR}^{\mathsf{FP}}$ *is an* X*-source.*

Definition 2.3 (Function privacy). *Let* $X \in \{(T,k)\text{-block}, (k_1, \ldots, k_T)\}$*. An identity-based encryption scheme* $\Pi = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *is* X-source function private *if for any probabilistic polynomial-time* X*-source function-privacy adversary* $\mathcal{A}$*, there exists a negligible function* $\nu(\lambda)$ *such that*

$$\mathbf{Adv}^{\mathsf{FP}}_{\Pi,\mathcal{A}}(\lambda) \stackrel{\mathsf{def}}{=} \left|\Pr\left[\mathsf{Expt}^{\mathsf{real}}_{\mathsf{FP},\Pi,\mathcal{A}}(\lambda) = 1\right] - \Pr\left[\mathsf{Expt}^{\mathsf{rand}}_{\mathsf{FP},\Pi,\mathcal{A}}(\lambda) = 1\right]\right| \leq \nu(\lambda),$$

where for each $\mathsf{mode} \in \{\mathsf{real}, \mathsf{rand}\}$ *and* $\lambda \in \mathbb{N}$ *the experiment* $\mathsf{Expt}^{\mathsf{mode}}_{\mathsf{FP},\Pi,\mathcal{A}}(\lambda)$ *is defined as follows:*

1. $(\mathrm{pp}, \mathrm{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$.
2. $b \leftarrow \mathcal{A}^{\mathsf{RoR}^{\mathsf{FP}}(\mathsf{mode},\mathrm{msk},\cdot),\mathsf{KeyGen}(\mathrm{msk},\cdot)}(1^\lambda, \mathrm{pp})$.
3. *Output* b.

In addition, such a scheme is statistically X*-source function private if the above holds for any* computationally-unbounded X*-source enhanced function-privacy adversary making a polynomial number of queries to the* $\mathsf{RoR}^{\mathsf{FP}}$ *oracle.*

Multi-shot vs. Single-Shot Adversaries. Note that Definition 2.3 considers adversaries that query the function-privacy oracle for any polynomial number of times. In fact, as adversaries are also given access to the key-generation oracle, this "multi-shot" definition is polynomially equivalent to its "single-shot" variant in which adversaries query the real-or-random function-privacy oracle $\mathsf{RoR}^{\mathsf{FP}}$ at most once. This is proved via a straightforward hybrid argument, where the hybrids are constructed such that only one query is forwarded to the function-privacy oracle, and all other queries are answered using the key-generation oracle.

Known Schemes That Are Not Function Private. To exercise our notion of function privacy we demonstrate that the anonymous IBE schemes of Boneh and Frankin [18], Gentry, Peikert and Vaikuntanathan [32], Agrawal, Boneh and Boyen [3], and Kurosawa and Phong [40] are not function private. We present simple and efficient attacks showing that the schemes [18,32] do not satisfy Definition 2.3, and note that almost identical attacks can be carried on [3,40]. As discussed in Section 1, other anonymous IBE schemes such as [31,22] for which we were not able to find such simple attacks, can always be *assumed* to be function private based on somewhat non-standard entropy-based assumptions (such assumptions would essentially state that the schemes satisfy our definition). In this paper we are interested in schemes whose function privacy can be based on standard assumptions.

The Boneh-Franklin scheme uses a random oracle $H : \mathcal{ID} \to \mathbb{G}$ and the secret key for id is $\mathrm{sk}_{\mathrm{id}} = H(\mathrm{id})^\alpha$ where $\alpha \leftarrow \mathbb{Z}_p$ is the master secret. The public parameters are g and $h = g^\alpha$ for some generator g of $\mathbb{G}$. Consider an

adversary that queries the real-or-random oracle with the circuit of the distribution that samples a uniformly distributed id for which the most significant bit of $\hat{e}(g^\alpha, H(\mathrm{id}))$ is 0. Clearly, this distribution has almost full entropy, and can be described by a circuit of polynomial size given the public parameters.[11] Then, given $\mathrm{sk}_{\mathrm{id}} = H(\mathrm{id})^\alpha$ the adversary outputs 0 if the most significant bit of $\hat{e}(g, \mathrm{sk}_{\mathrm{id}})$ is 0 and outputs 1 otherwise. Since $\hat{e}(g, \mathrm{sk}_{\mathrm{id}}) = \hat{e}(g^\alpha, H(\mathrm{id}))$ it is easy to see that the adversary has advantage $1/2$ in distinguishing the real mode from the rand mode, thereby breaking function privacy. In Section 1.1 we presented a modification of this scheme which is function private, and the reader is referred to the full version [19] for its proof of security.

In the scheme of Gentry, Peikert and Vaikuntanathan, the public parameters consist of a matrix $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}$ and the master secret key is a short basis for the lattice $\Lambda_q^\perp(\mathbf{A})$. A secret key corresponding to an identity id is a short vector $\mathbf{e} \in \mathbb{Z}^m$ such that $\mathbf{Ae} = H(\mathrm{id}) \in \mathbb{Z}_q^n$, where $H : \mathcal{ID} \to \mathbb{Z}_q^n$ is a random oracle. Consider an adversary that queries the real-or-random oracle with the circuit of the distribution that samples a uniformly distributed id for which the most significant bit of $H(\mathrm{id})$ is 0. Then, given $\mathrm{sk}_{\mathrm{id}} = \mathbf{e}$ the adversary outputs 0 if the most significant bit of $\mathbf{Ae}$ is 0 and outputs 1 otherwise. Since $\mathbf{Ae} = H(\mathrm{id})$ it is easy to see that the adversary has advantage $1/2$ in distinguishing the real mode from the rand mode, thereby breaking function privacy. In the full version [19] we present a modification of this scheme which is function private.

3 A Selectively-Secure DLIN-Based Scheme

In this section we present an IBE scheme based on the DLIN assumption in the standard model. For emphasizing the main ideas underlying our approach, we present here a *selectively* data private scheme, and refer the reader to for full version [19] for its extension to *full* data privacy. The scheme is based on the DLIN-based IBE of Kurosawa and Phong [40], which is an adaptation of the LWE-based IBE of Agrawal, Boneh and Boyen [3] to bilinear groups. The scheme is obtained by applying our "extract-augment-combine" approach, as discussed in Section 1.1.

The Scheme. Let $\mathsf{GroupGen}$ be a probabilistic polynomial-time algorithm that takes as input a security parameter 1^λ, and outputs $(\mathbb{G}, \mathbb{G}_{\mathrm{T}}, p, g, \hat{e})$ where $\mathbb{G}$ and $\mathbb{G}_{\mathrm{T}}$ are groups of prime order p, $\mathbb{G}$ is generated by g, p is a λ-bit prime number, and $\hat{e} : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_{\mathrm{T}}$ is a non-degenerate efficiently computable bilinear map. The scheme $\mathcal{IBE}_{\mathsf{DLIN1}} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ is parameterized by the security parameter $\lambda \in \mathbb{N}$. For any such $\lambda \in \mathbb{N}$, the scheme has parameters $m \geq 3$ and $\ell \geq 2$, identity space $\mathcal{ID}_\lambda = \mathbb{Z}_p^\ell$, and message space $\mathcal{M}_\lambda = \mathbb{G}_{\mathrm{T}}$.

- **Setup:** On input 1^λ sample $(\mathbb{G}, \mathbb{G}_{\mathrm{T}}, p, g, \hat{e}) \leftarrow \mathsf{GroupGen}(1^\lambda)$, $\mathbf{A}_0, \mathbf{A}_1, \ldots, \mathbf{A}_\ell, \mathbf{B} \leftarrow \mathbb{Z}_p^{2 \times m}$, and $\mathbf{u} \leftarrow \mathbb{Z}_p^2$. Output $\mathrm{pp} = \left(g, g^{\mathbf{A}_0}, g^{\mathbf{A}_1}, \ldots, g^{\mathbf{A}_\ell}, \mathbf{B}, g^{\mathbf{u}}\right)$ and $\mathrm{msk} = (\mathbf{A}_0, \mathbf{A}_1, \ldots, \mathbf{A}_\ell, \mathbf{u})$.

[11] More specifically, rejection sampling can be used to obtain a sufficiently good approximation.

– **Key generation:** On input the master secret key msk and an identity $\mathbf{id} = (\mathrm{id}_1, \ldots, \mathrm{id}_\ell) \in \mathbb{Z}_p^\ell$, sample $s_1, \ldots, s_\ell \leftarrow \mathbb{Z}_p$ and computes

$$\mathbf{F}_{\mathbf{id},(s_1,\ldots,s_\ell)} = \left[\mathbf{A}_0 \,\middle|\, \left(\sum_{i\in[\ell]} s_i \mathbf{A}_i \right) + \left(\sum_{i\in[\ell]} s_i \cdot \mathrm{id}_i \right) \mathbf{B} \right] \in \mathbb{Z}_p^{2\times 2m}.$$

Then, sample $\mathbf{v} \leftarrow \mathbb{Z}_p^{2m}$ such that $\mathbf{F}_{\mathbf{id},(s_1,\ldots,s_\ell)} \cdot \mathbf{v} = \mathbf{u} \pmod p$ and set $\mathbf{z} = g^{\mathbf{v}} \in \mathbb{G}^{2m}$. Outputs $\mathrm{sk}_{\mathbf{id}} = (s_1, \ldots, s_\ell, \mathbf{z})$.

– **Encryption:** On input the public parameters pp, an identity $\mathbf{id} = (\mathrm{id}_1, \ldots, \mathrm{id}_\ell) \in \mathbb{Z}_p^\ell$, and a message $\mathfrak{m} \in \mathbb{G}_\mathrm{T}$, sample $\mathbf{r} \leftarrow \mathbb{Z}_p^2$. Set $\mathbf{c}_0^\intercal = g^{\mathbf{r}^\intercal \mathbf{A}_0} \in \mathbb{G}^{1\times m}$, $\mathbf{c}_i^\intercal = g^{\mathbf{r}^\intercal [\mathbf{A}_i + \mathrm{id}_i \mathbf{B}]} \in \mathbb{G}^{1\times m}$ for all $i \in [\ell]$, $c_{\ell+1} = \hat{e}(g,g)^{\mathbf{r}^\intercal \mathbf{u}} \cdot \mathfrak{m} \in \mathbb{G}_\mathrm{T}$, and output $(\mathbf{c}_0, \mathbf{c}_1, \ldots, \mathbf{c}_\ell, c_{\ell+1}) \in \mathbb{G}^{(\ell+1)m} \times \mathbb{G}_\mathrm{T}$.

– **Decryption:** On input a ciphertext $c = (\mathbf{c}_0, \mathbf{c}_1, \ldots, \mathbf{c}_\ell, c_{\ell+1})$ and a secret key $\mathrm{sk} = (s_1, \ldots, s_\ell, \mathbf{z})$, output

$$\mathfrak{m} = c_{\ell+1} \cdot \hat{e} \left(\begin{bmatrix} \mathbf{c}_0 \\ \prod_{i\in[\ell]} \mathbf{c}_i^{s_i} \end{bmatrix}, \begin{vmatrix} \\ \mathbf{z} \\ \\ \end{vmatrix} \right)^{-1}.$$

Correctness. Note that

$$\mathbf{d}^\intercal = \left[\mathbf{c}_0^\intercal \,\middle|\, \prod_{i\in[\ell]} (\mathbf{c}_i^\intercal)^{s_i} \right] = g^{\mathbf{r}^\intercal \left[\mathbf{A}_0 \,\middle|\, \sum_{i\in[\ell]} s_i \mathbf{A}_i + \left(\sum_{i\in[\ell]} s_i \cdot \mathrm{id}_i\right)\mathbf{B} \right]} = g^{\mathbf{r}^\intercal \mathbf{F}_{\mathbf{id},(s_1,\ldots,s_\ell)}}.$$

We have $\hat{e}(\mathbf{d}, \mathbf{z}) = \hat{e}(g,g)^{\mathbf{r}^\intercal \mathbf{F}_{\mathbf{id},(s_1,\ldots,s_\ell)} \cdot \mathbf{v}} = \hat{e}(g,g)^{\mathbf{r}^\intercal \mathbf{u}}$. Therefore, dividing $c_{\ell+1}$ by $\hat{e}(\mathbf{d}, \mathbf{z})$ eliminates the term $\hat{e}(g,g)^{\mathbf{r}^\intercal \mathbf{u}}$ which recovers $\mathfrak{m}$ correctly.

Security. Due to space limitations we refer the reader to the full version [19] for the proof of the following theorem. Below we briefly highlight the main ideas underlying its proof.

Theorem 3.1. *The scheme $\mathcal{IBE}_{\mathsf{DLIN1}}$ is selectively data private based on the DLIN assumption, and is function private for:*

1. *(T,k)-block-sources for any $T = \mathrm{poly}(\lambda)$ and $k \geq \lambda + \omega(\log \lambda)$.*
2. *$(k_1, \ldots, k_T)$-sources for any $T = \mathrm{poly}(\lambda)$ and $(k_1, \ldots, k_T)$ such that $k_i \geq i \cdot \lambda + \omega(\log \lambda)$ for every $i \in [T]$.*

Proof Overview. The function privacy of the scheme follows quite naturally from our "extract" step, as discussed in Section 1.1. To prove selective data privacy under the DLIN assumption, given the challenge identity $\mathbf{id}^*$, we set up the public parameters $\{g^{\mathbf{A}_i}\}_{i\in[\ell]}$, $\mathbf{B}$, and $g^{\mathbf{u}}$ such that the matrix $\mathbf{G}_{\mathbf{id},\mathbf{s}} \stackrel{\mathsf{def}}{=} \left[\left(\sum_{i\in[\ell]} s_i \mathbf{A}_i \right) + \left(\sum_{i\in[\ell]} s_i \cdot \mathrm{id}_i \right) \mathbf{B} \right]$ is equipped with a 'punctured' trapdoor. This trapdoor allows us to sample a vector such that $\mathbf{F}_{\mathbf{id},\mathbf{s}} \cdot \mathbf{v} = \mathbf{u}$ whenever $\mathbf{G}_{\mathbf{id},\mathbf{s}}$

contains a non-zero scalar multiple of $\mathbf{B}$. This occurs whenever $\sum_{i\in[\ell]} s_i(\mathrm{id}_i - \mathrm{id}_i^*) \neq 0$. Thus, with all but a negligible probability, we can simulate the adversary's key-generation queries with specially chosen matrices as above.

To embed the DLIN challenge, the first two rows of the DLIN challenge are used to construct the public parameter $g^{\mathbf{A}_0}$. The third row of the challenge is either linearly dependent on the first two rows or chosen uniformly at random and independently. The third row of the challenge is embedded into the augmented challenge ciphertext that is either well-formed or uniform and independent of the adversary's view depending on the DLIN challenge. This is done by choosing secret matrices $\mathbf{R}_i^*$ and having $\mathbf{A}_i = \mathbf{A}_0\mathbf{R}_i^* - \mathrm{id}_i^*\mathbf{B}$. This generalizes the ideas of [3,40] to fit our "extract-augment-combine" approach and therefore provide function privacy.

4 Extensions and Open Problems

Our framework for function privacy yields a variety of extensions and open problems, both conceptual ones regarding our new notions, and technical ones regarding our specific approach and its resulting constructions. We now discuss several such extensions and open problems.

Chosen-Ciphertext Security. In terms of data privacy, in this paper we considered the standard notion of anonymity and message indistinguishability under an adaptive chosen-identity chosen-plaintext attack (known as anon-IND-ID-CPA). A natural extension of our results is to guarantee data privacy even against chosen-ciphertext attacks (known as anon-IND-ID-CCA). We note that our IBE schemes can be extended, using standard techniques, into two-level hierarchical IBE schemes that are anon-IND-ID-CPA-secure and their first level is function private. Then, by applying the generic transformation of Boneh, Canetti, Halevi and Katz [16], any such scheme can be used to construct an IBE scheme that is anon-IND-ID-CCA-secure and function private.

Applying Our Approach to other IBE Schemes. In Section 2 we presented simple attacks exemplifying that the anonymous IBE schemes presented in [18,32,3,40] are not function private. Nevertheless, we were able to rely on these schemes for designing new ones that are function private using our "extract-augment-combine" approach. For other anonymous IBE schemes, such as [31,22,11], we were not able to find attacks against their function privacy. An interesting open problem is to explore whether these schemes can be modified (possibly by applying our "extract-augment-combine" approach) to be function private based on standard assumptions. More generally, a natural open problem is to identify a specific property of identity-based encryption schemes that make them amenable to our "extract-augment-combine" approach.

Extension to Other Classes of Functions. As discussed in Section 1, in the general setting of functional encryption our schemes provide function privacy for the class of functions f_{id^*} defined as $f_{id^*}(id, m) = m$ if $id = id^*$, and $f_{id^*}(id, m) = \perp$ otherwise. A fascinating open problem is to construct schemes

that are function private for other classes of functions. A possible starting point is to consider function privacy for other, rather simple, functionalities, such as inner-product testing [39].

Robustness of Our Schemes. As pointed out by Abdalla, Bellare, and Neven [2], when using an anonymous IBE scheme as a public-key searchable encryption scheme [17,1], it is often desirable to use a "robust" IBE scheme: It should be difficult to produce a ciphertext that is valid for more than one identity. We note that our schemes do not satisfy such a notion of robustness. However, Abdalla et al. showed two generic transformations that transform any given IBE scheme into a robust one. In particular, these transformations can be applied to each of our schemes to make them robust (these transformations do not change the decryption keys, and thus function privacy is preserved). We leave it as an open problem to directly design function-private IBE schemes that are robust.

Acknowledgements. We thank the anonymous CRYPTO '13 reviewers for many useful comments. This work was supported by NSF, the DARPA PROCEED program, an AFO SR MURI award, a grant from ONR, an IARPA project provided via DoI/NBC, and by Samsung. Opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of DARPA or IARPA. Distrib. Statement "A:" Approved for Public Release, Distribution Unlimited.

References

1. Abdalla, M., Bellare, M., Catalano, D., Kiltz, E., Kohno, T., Lange, T., Malone-Lee, J., Neven, G., Paillier, P., Shi, H.: Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions. Journal of Cryptology 21(3), 350–391 (2008)
2. Abdalla, M., Bellare, M., Neven, G.: Robust encryption. In: Micciancio, D. (ed.) TCC 2010. LNCS, vol. 5978, pp. 480–497. Springer, Heidelberg (2010)
3. Agrawal, S., Boneh, D., Boyen, X.: Efficient lattice (H)IBE in the standard model. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 553–572. Springer, Heidelberg (2010)
4. Agrawal, S., Freeman, D.M., Vaikuntanathan, V.: Functional encryption for inner product predicates from learning with errors. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 21–40. Springer, Heidelberg (2011)
5. Agrawal, S., Gorbunov, S., Vaikuntanathan, V., Wee, H.: Functional encryption: New perspectives and lower bounds. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 500–518. Springer, Heidelberg (2013)
6. Baek, J., Safavi-Naini, R., Susilo, W.: Public key encryption with keyword search revisited. In: Gervasi, O., Murgante, B., Laganà, A., Taniar, D., Mun, Y., Gavrilova, M.L. (eds.) ICCSA 2008, Part I. LNCS, vol. 5072, pp. 1249–1259. Springer, Heidelberg (2008)
7. Barak, B., Goldreich, O., Impagliazzo, R., Rudich, S., Sahai, A., Vadhan, S.P., Yang, K.: On the (im)possibility of obfuscating programs. Journal of the ACM 59(2), 6 (2012)

8. Bellare, M., Boldyreva, A., O'Neill, A.: Deterministic and efficiently searchable encryption. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 535–552. Springer, Heidelberg (2007)
9. Bellare, M., Brakerski, Z., Naor, M., Ristenpart, T., Segev, G., Shacham, H., Yilek, S.: Hedged public-key encryption: How to protect against bad randomness. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 232–249. Springer, Heidelberg (2009)
10. Bellare, M., Fischlin, M., O'Neill, A., Ristenpart, T.: Deterministic encryption: Definitional equivalences and constructions without random oracles. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 360–378. Springer, Heidelberg (2008)
11. Bellare, M., Kiltz, E., Peikert, C., Waters, B.: Identity-based (Lossy) trapdoor functions and applications. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 228–245. Springer, Heidelberg (2012)
12. Bellare, M., O'Neill, A.: Semantically-secure functional encryption: Possibility results, impossibility results and the quest for a general definition. Cryptology ePrint Archive, Report 2012/515 (2012)
13. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: Proceedings of the 1st ACM Conference on Computer and Communications Security, pp. 62–73 (1993)
14. Bethencourt, J., Song, D., Waters, B.: New techniques for private stream searching. ACM Transactions on Information and System Security 12(3) (2009)
15. Boldyreva, A., Fehr, S., O'Neill, A.: On notions of security for deterministic encryption, and efficient constructions without random oracles. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 335–359. Springer, Heidelberg (2008)
16. Boneh, D., Canetti, R., Halevi, S., Katz, J.: Chosen-ciphertext security from identity-based encryption. SIAM Journal on Computing 36(5), 1301–1328 (2007)
17. Boneh, D., Di Crescenzo, G., Ostrovsky, R., Persiano, G.: Public key encryption with keyword search. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 506–522. Springer, Heidelberg (2004)
18. Boneh, D., Franklin, M.K.: Identity-based encryption from the Weil pairing. SIAM Journal on Computing 32(3), 586–615 (2003); Preliminary version in Kilian, J. (ed.): CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (2001)
19. Boneh, D., Raghunathan, A., Segev, G.: Function-private identity-based encryption: Hiding the function in functional encryption. Cryptology ePrint Archive, Report 2013/283 (2013)
20. Boneh, D., Sahai, A., Waters, B.: Functional encryption: Definitions and challenges. In: Ishai, Y. (ed.) TCC 2011. LNCS, vol. 6597, pp. 253–273. Springer, Heidelberg (2011)
21. Boneh, D., Waters, B.: Conjunctive, subset, and range queries on encrypted data. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 535–554. Springer, Heidelberg (2007)
22. Boyen, X., Waters, B.: Anonymous hierarchical identity-based encryption (Without random oracles). In: Dwork, C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 290–307. Springer, Heidelberg (2006)
23. Brakerski, Z., Segev, G.: Better security for deterministic public-key encryption: The auxiliary-input setting. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 543–560. Springer, Heidelberg (2011)
24. Camenisch, J., Kohlweiss, M., Rial, A., Sheedy, C.: Blind and anonymous identity-based encryption and authorised private searches on public key encrypted data. In: Jarecki, S., Tsudik, G. (eds.) PKC 2009. LNCS, vol. 5443, pp. 196–214. Springer, Heidelberg (2009)

25. Canetti, R.: Towards realizing random oracles: Hash functions that hide all partial information. In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 455–469. Springer, Heidelberg (1997)
26. Canetti, R., Kalai, Y.T., Varia, M., Wichs, D.: On symmetric encryption and point obfuscation. In: Micciancio, D. (ed.) TCC 2010. LNCS, vol. 5978, pp. 52–71. Springer, Heidelberg (2010)
27. Chang, Y.-C., Mitzenmacher, M.: Privacy preserving keyword searches on remote encrypted data. In: Ioannidis, J., Keromytis, A.D., Yung, M. (eds.) ACNS 2005. LNCS, vol. 3531, pp. 442–455. Springer, Heidelberg (2005)
28. Chase, M., Kamara, S.: Structured encryption and controlled disclosure. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 577–594. Springer, Heidelberg (2010)
29. Curtmola, R., Garay, J.A., Kamara, S., Ostrovsky, R.: Searchable symmetric encryption: Improved definitions and efficient constructions. Journal of Computer Security 19(5), 895–934 (2011)
30. Fuller, B., O'Neill, A., Reyzin, L.: A unified approach to deterministic encryption: New constructions and a connection to computational entropy. In: Cramer, R. (ed.) TCC 2012. LNCS, vol. 7194, pp. 582–599. Springer, Heidelberg (2012)
31. Gentry, C.: Practical identity-based encryption without random oracles. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 445–464. Springer, Heidelberg (2006)
32. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Proceedings of the 40th Annual ACM Symposium on Theory of Computing, pp. 197–206 (2008)
33. Goldreich, O., Ostrovsky, R.: Software protection and simulation on oblivious rams. Journal of the ACM 43(3), 431–473 (1996)
34. Goldwasser, S., Kalai, Y.T., Popa, R.A., Vaikuntanathan, V., Zeldovich, N.: Reusable garbled circuits and succinct functional encryption. In: Proceedings of the 45th Annual ACM Symposium on Theory of Computing (to appear, 2013)
35. Goldwasser, S., Kalai, Y.T.: On the impossibility of obfuscation with auxiliary input. In: Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science, pp. 553–562 (2005)
36. Golle, P., Staddon, J., Waters, B.: Secure conjunctive keyword search over encrypted data. In: Jakobsson, M., Yung, M., Zhou, J. (eds.) ACNS 2004. LNCS, vol. 3089, pp. 31–45. Springer, Heidelberg (2004)
37. Gorbunov, S., Vaikuntanathan, V., Wee, H.: Functional encryption with bounded collusions via multi-party computation. In: Safavi-Naini, R., Canett, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 162–179. Springer, Heidelberg (2012)
38. Kamara, S., Papamanthou, C., Roeder, T.: Dynamic searchable symmetric encryption. In: ACM Conference on Computer and Communications Security, pp. 965–976 (2012)
39. Katz, J., Sahai, A., Waters, B.: Predicate encryption supporting disjunctions, polynomial equations, and inner products. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 146–162. Springer, Heidelberg (2008)
40. Kurosawa, K., Phong, L.T.: Maximum leakage resilient IBE and IPE. Cryptology ePrint Archive, Report 2011/628 (2011)
41. Lynn, B., Prabhakaran, M., Sahai, A.: Positive results and techniques for obfuscation. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 20–39. Springer, Heidelberg (2004)

42. Mironov, I., Pandey, O., Reingold, O., Segev, G.: Incremental deterministic public-key encryption. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 628–644. Springer, Heidelberg (2012)
43. O'Neill, A.: Definitional issues in functional encryption. IACR Cryptology ePrint Archive, Report 2010/556 (2010)
44. Ostrovsky, R., Skeith III., W.E.: Private searching on streaming data. Journal of Cryptology 20(4), 397–430 (2007)
45. Raghunathan, A., Segev, G., Vadhan, S.: Deterministic public-key encryption for adaptively chosen plaintext distributions. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 93–110. Springer, Heidelberg (2013)
46. Shen, E., Shi, E., Waters, B.: Predicate privacy in encryption systems. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 457–473. Springer, Heidelberg (2009)
47. Shi, E., Bethencourt, J., Chan, H.T.-H., Song, D., Perrig, A.: Multi-dimensional range query over encrypted data. In: IEEE Symposium on Security and Privacy, pp. 350–364 (2007)
48. Song, D.X., Wagner, D., Perrig, A.: Practical techniques for searches on encrypted data. In: IEEE Symposium on Security and Privacy, pp. 44–55 (2000)
49. van Liesdonk, P., Sedghi, S., Doumen, J., Hartel, P., Jonker, W.: Computationally efficient searchable symmetric encryption. In: Jonker, W., Petković, M. (eds.) SDM 2010. LNCS, vol. 6358, pp. 87–100. Springer, Heidelberg (2010)
50. Wee, H.: On obfuscating point functions. In: Proceedings of the 37th Annual ACM Symposium on Theory of Computing, pp. 523–532 (2005)
51. Wee, H.: Dual projective hashing and its applications — lossy trapdoor functions and more. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 246–262. Springer, Heidelberg (2012)