Anonymity in Public-Key Encryption

Master's Thesis[1]

Irippuge Milinda Perera

The Graduate Center of CUNY
ipereara@gc.cuny.edu

May 9, 2013

[1] This thesis was submitted in partial fulfillment of the requirements for the degree of Master of Science from the Department of Computer Science at the City College of CUNY.

Abstract

The basic security property provided by a public-key encryption scheme is *data secrecy*, a guarantee that the encrypted messages (or ciphertexts) do not leak any information regarding the original content. The ciphertexts, however, may still reveal the identities of the intended recipients. This is unacceptable in a world where threats to personal privacy are increasing rapidly. As a step toward solving the above problem, the cryptographic community has recently proposed various anonymous encryption schemes. In addition to data secrecy, these schemes also provide *key privacy* or *anonymity*, i.e., they also guarantee that ciphertexts do not reveal any information regarding the identities of the recipients. In this survey, we review anonymous schemes in several application domains of public-key cryptography, namely standard public-key encryption, identity-based encryption, and broadcast encryption.

Keywords: Key Privacy, Recipient Anonymity, Public-Key Encryption, Identity-Based Encryption, Broadcast Encryption.

Contents

List of Tables

1 Introduction

Today we are living in a world where we rely on digital technologies more than ever before. Email, social media, online banking, and online shopping are only a few examples of how we have incorporated these digital technologies into our day-to-day lives. Cryptography has played a very significant role in providing us with various security guarantees such as authenticity, confidentiality, and privacy that we require when using these digital technologies. The basic security guarantees that the cryptographic systems provide can be categorized into two varieties: data secrecy and key privacy (a.k.a. anonymity). In this review paper, we are interested in surveying the intricacies of key privacy in several mainstream cryptographic settings.

Historically, cryptography was considered the art of writing or solving codes. As a result, proving the advertised security guarantees of the early cryptographic schemes (such as Caesar cipher, Vigenère cipher, Enigma machine, etc.) was never an integral part of their design [43]. And as a consequence, all these early ciphers have been badly broken[1]. Clearly, for the security purposes of the modern digital communications, seeing cryptography as an art is not enough.

The first step in treating cryptography as a science was taken by Claude Shannon [56] in his famous paper "Communication Theory of Secrecy Systems", where he presented the first rigorous mathematical proof of the Vernam cipher (a.k.a. one-time pad). Then in 1976, Diffie and Hellman [24] initiated the field of *public-key cryptography* with their remarkable paper "New Directions in Cryptography". Since then, there has been a multitude of advances such as new cryptographic protocols, stronger security guarantees, etc. in the cryptographic research community.

Among the basic application domains of public-key cryptography are standard public-key encryption, identity-based encryption, and broadcast encryption. In this review paper, we are examining how the notion of anonymity appears in these three domains. It should be noted that special attention is given to the area of broadcast encryption because of our recent work in that setting and our interest to do further research in that direction.

Public-key encryption, which deals with secure point-to-point communication of messages, is undoubtedly the most popular application domain of public-key cryptography. The basic security guarantee provided by public-key encryption schemes is data secrecy (i.e., the encrypted message or the ciphertext does not reveal any information regarding the original message). The notion of anonymity in public-key encryption additionally requires that a ciphertext does not reveal any information regarding the public key under which the ciphertext is generated. In Section 3, we formally review what public-key encryption is and what key privacy in public-key encryption means, and also compare the efficiency parameters of the existing constructions of public-key encryption with key privacy guarantees.

The public key of a user in a public-key encryption scheme is usually a very long bit-string with a specific structure. And as a result, it is usually hard to remember. In order to solve this problem, Shamir [55] proposed the notion of identity-based encryption in 1984. An identity-based encryption scheme is a public-key encryption scheme with the relaxation that the public key of a user can be an arbitrary bit-string (such as an email address). A generalization of identity-based encryption is hierarchical identity-based encryption where the identities of the users are distributed in a hierarchical structure. The basic security guarantee provided by (hierarchical) identity-based encryption is also data secrecy. The notion of anonymity in this setting is an extension from the

[1] Here we mean that a cipher is broken when it is possible with overwhelming advantage for an adversary to recover the message embedded in a ciphertext without knowing the secret key.

setting of public-key encryption. In Section 4 of this paper, we formally explain what the setting of (hierarchical) identity-based encryption is and what the notion of anonymity in this setting means, and also compare the parameters of currently available constructions of (hierarchical) identity-based encryption with anonymity properties.

Both public-key encryption and identity-based encryption provide the means for secret communication of data in point-to-point communication. The setting of broadcast encryption, instead, deals with secure *multicast* communication of data. Although the notion of anonymity also appears in this setting, its definition is not as trivial as in the other two application domains mentioned earlier. This is due to the complexities that arise when creating a single ciphertext to a group of users. In Section 5 of this paper, we formally explain the setting of broadcast encryption and the notions of anonymity that arise in this setting. As we will see in this section, there are two notions of anonymity in broadcast encryption namely, full-anonymity and outsider-anonymity. After explaining the differences between these two notions, we also present a thorough analysis of the existing anonymous broadcast encryption constructions in the same section.

Organization. Section 2 provides some background knowledge on the notation and the cryptographic assumptions that are referred to in the remaining sections of this review paper. Also included in Section 2 is a review of the subset cover framework of [50]. Section 3 is about the notion of key privacy in public-key encryption. Section 4 reviews the notion of anonymity in (hierarchical) identity-based encryption. Section 5, which is the bulk of this review paper, is about the notion of anonymity in broadcast encryption. Finally, in Section 6, we conclude this paper.

2 Background

2.1 Notation

We denote by $\|$ the string concatenation operation. For a given vector $\vec{a}$ and an element b, we denote by $\vec{a} : b$ the vector obtained by appending b at the end of the vector $\vec{a}$. We use $\mathsf{Prfx}(\vec{a})$ to denote the set of all prefix vectors of $\vec{a}$ with non-zero length. We use $x \leftarrow\!\$\ X$ to denote that variable x is chosen uniformly at random from set X.

2.2 Diffie-Hellman Assumptions

Let $\mathbb{G} = \langle g \rangle$ be a group with a generator g and order q. Define the function $\mathsf{dh} : \mathbb{G}^2 \to \mathbb{G}$ as

$$\mathsf{dh}(X, Y) := Z, \text{ where } X = g^x, Y = g^y, \text{ and } Z = g^{xy} \text{ for } x, y \in \mathbb{Z}_q.$$

Computational Diffie-Hellman (CDH) [24]. We say that the CDH problem is (t, ϵ)-hard relative to $\mathbb{G}$ if for all t-time adversaries $\mathcal{A}$ we have

$$\Big|\Pr\big[\mathcal{A}(\mathbb{G}, q, g, g^x, g^y) = \mathsf{dh}(g^x, g^y)\big]\Big| \leq \epsilon,$$

where $x, y \leftarrow\!\$\ \mathbb{Z}_q$ and the probability is over the random coins used to generate x, y and by $\mathcal{A}$.

Decisional Diffie-Hellman (DDH) [24]. We say that the DDH problem is (t, ϵ)-hard relative to $\mathbb{G}$ if for all t-time adversaries $\mathcal{A}$ we have

$$\Big|\Pr\big[\mathcal{A}(\mathbb{G}, q, g, g^x, g^y, g^{xy}) = 1\big] - \Pr\big[\mathcal{A}(\mathbb{G}, q, g, g^x, g^y, g^z) = 1\big]\Big| \leq \epsilon,$$

where $x, y, z \leftarrow\$ \mathbb{Z}_q$ and the probability is over the random coins used to generate x, y, z and by $\mathcal{A}$.

Strong Twin Computational Diffie-Hellman (s2CDH) [20]. Define $\mathsf{2dh} : \mathbb{G}^3 \to \mathbb{G}^2$ as

$$\mathsf{2dh}(X_1, X_2, Y) \coloneqq (\mathsf{dh}(X_1, Y), \mathsf{dh}(X_2, Y)).$$

For fixed $X_1, X_2 \in \mathbb{G}$, also define the predicate $\mathsf{2dhp} : \mathbb{G}^3 \to \{\mathsf{True}, \mathsf{False}\}$ as

$$\mathsf{2dhp}(X_1, X_2, \hat{Y}, \hat{Z}_1, \hat{Z}_2) \coloneqq \mathsf{2dh}(X_1, X_2, \hat{Y}) \overset{?}{=} (\hat{Z}_1, \hat{Z}_2).$$

We say that the s2CDH problem is (t, ϵ)-hard relative to $\mathbb{G}$ if for all t-time adversaries $\mathcal{A}$ we have

$$\left| \Pr[\mathcal{A}^{\mathsf{2dhp}(g^{x_1}, g^{x_2}, \cdot, \cdot, \cdot)}(g^{x_1}, g^{x_2}, g^y) = \mathsf{2dh}(g^{x_1}, g^{x_2}, g^y)] \right| \leq \epsilon,$$

where $x_1, x_2, y \leftarrow\$ \mathbb{Z}_q$ and the probability is over the randomness used to sample x_1, x_2, y and by $\mathcal{A}$.

Remark 2.1. As shown in [20], the s2CDH assumption is equivalent to the CDH assumption.

2.3 Review of the Subset Cover Framework

The subset cover (SC) framework proposed by Naor et al. [50] is a system that abstracts a variety of revocation schemes in the symmetric-key setting where only the Center can broadcast. In a nutshell, a revocation scheme belonging to the SC framework defines a collection of subsets $\mathcal{S}$ of the universe of users $U = [1, N]$ in the system. During the key generation phase, the Center assigns to each subset $S_i \in \mathcal{S}$ a long-lived key k_i, which is also given to each user belonging to S_i. When the Center wants to broadcast a message m, it generates a short-lived session key $\widehat{k}$, determines the set of revoked users R, finds a set of disjoint subsets $\widehat{\mathcal{S}}$ from $\mathcal{S}$ that contains or "covers" all the users in $U \backslash R$, encrypts $\widehat{k}$ using the long-lived keys corresponding to the subsets in $\widehat{\mathcal{S}}$, and finally broadcasts the encryption of m under $\widehat{k}$ and the encryptions of $\widehat{k}$ to all the users in the system. Upon receiving a broadcast ciphertext, a user can decrypt successfully and obtain m if and only if that user is part of the authorized set (i.e., the user possesses a long-lived key corresponding to some subset of $\widehat{\mathcal{S}}$).

The authors in [50] also presented two concrete revocation schemes, namely the *complete subtree* (CS) method and the *subset difference* (SD) method. In the CS method, which is the simplest of the two, the ciphertext length is $O(r \log(\frac{N}{r}))$ and the private key length at a receiver is $O(\log N)$, where r is the number of revoked users. In the SD method, the one with more involved computations, the ciphertext length reduces to $O(r)$ while the private key length increases to $O(\log^2 N)$. Another crucial difference between the two schemes is that the assignment of the long-lived keys in the former is information-theoretic, whereas in the latter its computational. Below we provide a short description of the CS method, and we refer the reader to [50] for further details on the SD method.

Complete Subtree Method. In this scheme, the N users are represented as the leaves of a perfect binary tree $\mathcal{T}$ and the collection of subsets $\mathcal{S}$ contains all possible subtrees of $\mathcal{T}$. In case N is not a power of 2, some dummy users are added to the system. During the key generation phase, every subtree in $\mathcal{S}$ is assigned a long-lived symmetric key which is also made available to all the users belonging to that subtree. Since every user is a member of all the subtrees rooted at each node in the path from the root of $\mathcal{T}$ down to the leaf corresponding to that user, the secret key length at a user is $O(\log N)$. The ciphertext length becomes $O(r \log(\frac{N}{r}))$ due to the fact that it requires on average a logarithmic number of subtrees to revoke r users (see [50] for a formal analysis).

Extension of the SC Framework to the Public-Key Setting. The original SC framework was defined for the symmetric-key setting. In [25], Dodis and Fazio extended the SC framework to the public-key setting by combining a novel assignment of hierarchical identifiers (HIDs) to the nodes in $\mathcal{T}$ with (hierarchical) identity-based encryption ((H)IBE). For completeness, we only explain below the extension of the CS method. We refer to [25] for the specifics regarding the SD method.

The assignment of HIDs to the nodes in $\mathcal{T}$ goes as follows. First, the root of $\mathcal{T}$ is assigned a special identifier denoted by ε. Next, each edge e of $\mathcal{T}$ is assigned the identifier $\mathsf{ID}_e \in \{0, 1\}$ depending on whether the edge connects to the left child or to the right child. Then, the hierarchical identifier HID_v of any node v can be computed by concatenating all the identifiers starting from the root of $\mathcal{T}$ down to v (i.e., $\mathsf{HID}_v := \varepsilon \| \mathsf{ID}_{e_1} \| \ldots \| \mathsf{ID}_{e_{\log N}}$). It is important to note that any prefix of HID_v represents a valid HID of a parent node of v.

Once the HIDs of the nodes are assigned, the authors employ an IBE scheme in order to encrypt the short-lived session keys during broadcasts. The long-lived keys of the subsets in $\mathcal{S}$ now become the IBE keys corresponding to the HIDs of the nodes in $\mathcal{T}$. Since the structure of the $\mathcal{T}$ and the assignment of HIDs are publicly known to all the users, any user in the system can be a sender as well as a receiver. In the public-key setting, the Center becomes the trusted authority that provides each user with the required IBE keys.

3 Key Privacy in Public-Key Encryption

Public-key Encryption. At the heart of modern cryptography is public-key encryption in which each participant has a pair of keys: a *public key* and a related *secret key*. The public key can be used by anyone to encrypt a message and obtain a ciphertext that can be decrypted only by the holder of the corresponding secret key. The pioneers of seeing cryptography in this *asymmetric* way are Diffie and Hellman [24], and also Merkle [49]. Although Diffie and Hellman introduced the notion of public-key encryption, the first concrete scheme was proposed by Rivest, Shamir, and Adleman [52] based on the hardness of their famous RSA problem. Later, in 1985, El Gamal [32] presented an encryption scheme by slightly modifying the key-exchange protocol of Diffie and Hellman [24]. In other words, Diffie and Hellman was very close to building the first public-key encryption scheme.

The basic security requirement of a public-key encryption scheme is the secrecy of the encrypted data. In summary, the idea is that an adversary, knowing the messages m_0, m_1 and given a ciphertext c containing either m_0 or m_1 encrypted under the *same* public key, is unable to identify which message is embedded in c without knowing the corresponding secret key. This notion of data secrecy has appeared in formalizations such as semantic security [40] or non-malleability [29]. A survey of the relations among these and other notions of security related to data secrecy can be found in [7].

Key-Private Public-Key Encryption. Another security requirement of a public-key encryption scheme, the one we are interested in this paper, is the indistinguishability of the public keys used to generate ciphertexts. This notion was originally formalized by Bellare et al. [6] where the authors coined the term "key privacy". In a nutshell, this requirement states that an adversary, knowing the two public keys pk_0, pk_1 and having a ciphertext c generated under either pk_0 or pk_1, has negligible advantage in determining under which public key c is generated. In other words, the ciphertext should not leak any information regarding its corresponding public key.

There are numerous examples motivating the need for key-private public-key encryption schemes. One such example is in the setting of mobile communication networks. Assume a group of users want to exchange sensitive data through a public WiFi network and also want to remain anonymous in the eyes of a third party. Also assume that the owner of the base station is honest but curious (i.e., it attempts to "unlawfully" gather information regarding the users while following the communication protocol correctly). If the users employ an encryption scheme only with data secrecy guarantees, the base station owner could do traffic analysis on their ciphertexts and obtain their identities. However, if the users employ an encryption scheme that additionally provides key privacy, they can prevent the base station from extracting their identities from the ciphertexts.

As the authors in [6] noticed, it would be ideal to prove that the existing schemes possess the key privacy guarantees without having to construct new schemes from the ground up. Unfortunately, a public-key encryption scheme could meet the strongest notion of data secrecy and have no guarantee of key privacy. As a pedagogical example, the ciphertext could have the corresponding public key attached to it. Since the public key is already known by everyone, a data-secrecy adversary gains no advantage by seeing the attached public key. Therefore, the existing public-key encryption schemes meeting only data secrecy requirements must be re-analyzed for key privacy requirements.

That is exactly what Bellare et al. did in [6] after formally defining the security model of key-private public-key encryption. Specifically, the authors analyzed discrete log-based schemes such as El Gamal and Cramer-Shoup, and RSA-based schemes such as RSA-OAEP, and reported on whether these schemes meet the key privacy requirements. We will provide a survey of these results in Section 3.3.

3.1 Setting

Definition 3.1: A public-key encryption scheme, associated with a message space $\mathcal{MSP}$ and a ciphertext space $\mathcal{CSP}$, is a tuple of probabilistic polynomial-time (PPT) algorithms (KeyGen, $\mathsf{Encrypt}$, $\mathsf{Decrypt}$) such that:

$\boldsymbol{(pk, sk)} \leftarrow \textbf{KeyGen}\boldsymbol{(1^\lambda)}$**:** KeyGen takes the security parameter 1^λ as input and outputs a public/secret key pair (pk, sk).

$\boldsymbol{c} \leftarrow \textbf{Encrypt}\boldsymbol{(pk, m)}$**:** $\mathsf{Encrypt}$ takes the public key pk and a message $m \in \mathcal{MSP}$ as inputs and outputs a ciphertext $c \in \mathcal{CSP}$.

$\boldsymbol{m/\perp} := \textbf{Decrypt}\boldsymbol{(sk, c)}$**:** Given a secret key sk and a ciphertext $c \in \mathcal{CSP}$, $\mathsf{Decrypt}$ either outputs a message $m \in \mathcal{MSP}$ or the failure symbol $\perp$. $\mathsf{Decrypt}$ is assumed to be deterministic.

Correctness. For every $m \in \mathcal{MSP}$, if (pk, sk) is output by $\mathsf{KeyGen}(1^\lambda)$, then $\mathsf{Decrypt}(sk, \mathsf{Encrypt}(pk, m)) = m$. $\lozenge$

3.2 Security Models

Now we present the formal definitions of key privacy in public-key encryption. Following the approach used by Bellare et al. [6], we present these definitions as games played between a PPT challenger and an adversary. First, we present the IK-IND-CCA game corresponding to the chosen ciphertext attack of key privacy. Next, we show how to obtain the IK-IND-CPA game corresponding to the chosen plaintext attack of key privacy by slightly modifying the IK-IND-CCA game.

Definition 3.2: For a given key-private public-key encryption scheme, the IK-IND-CCA game played between a PPT challenger $\mathcal{C}$ and an adversary $\mathcal{A}$, is defined as:

Table 1: Comparison of the parameters of key-private public-key encryption schemes.

Scheme	**Key-Private**	**Attack**	**ROM**
El Gamal [32]	Yes	CPA	No
Cramer-Shoup [22]	Yes	CCA	No
RSA-Plain [52]	No	–	–
RSA-OAEP* [6]	Yes	CCA	Yes

Setup: $\mathcal{C}$ runs $(pk_0, sk_0) \leftarrow \mathsf{KeyGen}(1^\lambda)$, $(pk_1, sk_1) \leftarrow \mathsf{KeyGen}(1^\lambda)$. It gives $\mathcal{A}$ the resulting public keys and keeps the secret keys to itself.

Phase 1: $\mathcal{A}$ adaptively issues queries $q_1, \ldots, q_m$ where each q_i is the following:

- Decryption query (b', c): $\mathcal{A}$ issues a decryption query on the public key $pk_{b'}$ for $b' \in \{0,1\}$ and a ciphertext c. $\mathcal{C}$ computes $\mathsf{Decrypt}(sk_{b'}, c)$ and gives the result to $\mathcal{A}$.

Challenge: $\mathcal{A}$ gives $\mathcal{C}$ two *equal* length messages $m_0^*, m_1^* \in \mathcal{MSP}$. $\mathcal{C}$ picks $b^* \leftarrow\$ \{0,1\}$, computes $c^* \leftarrow \mathsf{Encrypt}(pk_{b^*}, m_{b^*}^*)$, and sends c^* to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ adaptively issues additional queries $q_{m+1}, \ldots, q_n$ where each q_i is one of the following:

- Decryption query (b', c) such that $c \neq c^*$.

$\mathcal{C}$ replies as in *Phase 1*.

Guess: $\mathcal{A}$ outputs a guess $b \in \{0,1\}$ and wins if $b = b^*$.

The adversary in this game is called an IK-IND-CCA adversary and its advantage is defined as

$$\mathsf{Adv}_{\mathcal{A},\Pi}^{\text{IK-IND-CCA}} := \left|\Pr[b = b^*] - \tfrac{1}{2}\right|,$$

where the probability is over the random coins used by $\mathcal{C}$ and $\mathcal{A}$. $\lozenge$

Definition 3.3: A key-private public-key encryption scheme Π is (t, Q_d, ϵ)-IK-CCA-secure if for any t-time IK-IND-CCA adversary making at most Q_d adaptive decryption queries we have $\mathsf{Adv}_{\mathcal{A},\Pi}^{\text{IK-IND-CCA}} \leq \epsilon$. $\lozenge$

By restricting the adversary from submitting any decryption queries in *Phase 1* and *Phase 2* of the IK-IND-CCA game, we obtain the IK-IND-CPA game.

Definition 3.4: A key-private public-key encryption scheme Π is (t, ϵ)-IK-CPA-secure if Π is $(t, 0, \epsilon)$-IK-CCA-secure. $\lozenge$

3.3 Constructions

In this section, we review the key privacy guarantees of the most famous public-key encryption schemes in the literature. These results were obtained from the analysis carried out by Bellare et al. in [6]. Table 1 reports a high-level comparison of these schemes.

As you can see, the authors in [6] have been able to prove the key privacy of the discrete log-based schemes (ElGamal and Cramer-Shoup). They have done this using the same computational assumptions used to establish data secrecy of these schemes. Specifically, the assumption is that the DDH problem is hard in the underlying group which is shared by all the users in the system.

In the case of the plain RSA encryption, the authors notice the following simple attack that gives even an IK-IND-CPA adversary a non-negligible advantage in breaking the key privacy of the cryptosystem. As we know, the RSA ciphertext is generated as $c = m^e \bmod N$ where m, the message, is an element of $\mathbb{Z}_N^*$ and e is the public exponent. Suppose the adversary is given the two public keys $pk_0 = (N_0, e_0)$, $pk_1 = (N_1, e_1)$. Without any loss of generality, assume that $N_0 < N_1$. Now, given a ciphertext c, the adversary does the following. If $c > N_0$ it says that c was generated under the public key pk_1, otherwise, it says that c was created under the public key pk_0. It is easy to notice that the adversary's advantage is *not* negligible. Thus, the plain RSA scheme does not provide any key privacy guarantees.

Next, the authors consider the key privacy of the currently used variant of RSA-based encryption standard: RSA-OAEP [8]. The data secrecy of RSA-OAEP under the chosen ciphertext attack is proven secure in the random oracle model. Unfortunately, as the authors in [6] notice, RSA-OAEP does not provide key privacy. In order to overcome this problem, they propose a variant of RSA-OAEP (RSA-OAEP* in Table 1) that can be proven key-private also in the random oracle under the assumption that RSA is a one-way function. The computational cost of $\mathsf{Encrypt}$ operation in RSA-OAEP* is twice as expensive as RSA-OAEP. However, the cost of $\mathsf{Decrypt}$ operation is the same. Therefore, RSA-OAEP* seems to be an attractive public-key encryption scheme for the systems that are already using RSA-OAEP and also requiring key privacy guarantees.

4 Anonymity in (Hierarchical) Identity-Based Encryption

Identity-Based Encryption (IBE). Identity-based encryption is a public-key encryption scheme where the public key of a user can be an arbitrary bit-string such as an email address. A central key generation authority possessing a master secret key produces a secret key corresponding to a given identity. IBE greatly simplifies the problem of key distribution and management in public-key encryption since now the users don't have to worry about exchanging long and unintelligible public keys. Although this concept was suggested by Shamir in 1984 [55], an efficient and provably secure construction was not proposed until the work of Boneh and Franklin in 2001 [12]. Since then, there have been several IBE constructions such as [9, 13, 58] to name a few.

Hierarchical Identity-Based Encryption (HIBE). Since having a single key generation authority is undesirable due to the computational overhead in a large network of users, the notion of hierarchical identity-based encryption was proposed in [38, 42]. HIBE is a generalized version of IBE that allows delegation of keys in a hierarchical structure. At the top of this hierarchy is the central key generation authority. At the following level are several sub-authorities. Each of these sub-authorities holds a delegation key that allows it to decrypt the ciphertexts destined to it as well as to the users below it in the hierarchy. Each user belonging to a sub-authority is also allowed to extend the hierarchy by becoming a sub-sub-authority. Since the introduction, there have been several HIBE constructions in the literature (e.g., [10, 37, 46, 48, 58]).

Anonymous (Hierarchical) Identity-Based Encryption (A(H)IBE). The notion of anonymity in identity-based encryption is a direct extension of the notion of key privacy in public-key encryption [6]. Specifically, an A(H)IBE scheme is anonymous if its ciphertexts do not leak the identities of the recipients. This concept was originally proposed by Abdalla et al. in [2] where they investigated public-key encryption with keyword search (PEKS) [11].

The main application of A(H)IBE schemes is in anonymous communication systems [31]. As we noted earlier in Section 3, key-private public-key encryption schemes allow the senders to prevent an adversary from extracting their identities from the ciphertexts they send. However, since the users have to obtain the public keys of the receiving parties in the clear, the adversary is still capable of performing some form of traffic analysis on this exchange of public keys. In contrast, A(H)IBE schemes are resilient to even this type of attackers since a sender in an A(H)IBE schemes is not required to retrieve the recipient's public key, which is a simple bit-string, from a public location.

Another interesting application of A(H)IBE appears in PEKS as Abdalla et al. noticed in [2]. PEKS is a system where each ciphertext is associated with a keyword with the requirement that the ciphertext does not leak any information regarding the keyword. A user is given along with his decryption key a trapdoor for each keyword he is authorized to use. Now, given a batch of ciphertexts (e.g., stored in a remote database), a user can delegate the task of finding the ciphertexts associated to one of his keywords to an honest but curious third party (e.g., database administrator) by giving that third party the corresponding trapdoor. Then, the third party can find all the ciphertexts associated with the keyword corresponding to the given trapdoor without knowing what the keyword is or which messages are encrypted in the ciphertexts. In [2], Abdalla et al. presented a generic constriction that built a PEKS scheme by using an AIBE scheme as an underlying primitive.

Although the authors in [2] introduced the notion of A(H)IBE, they didn't provide any concrete constructions. They did, however, notice that the very first IBE scheme by Boneh and Franklin [12] was indeed anonymous in the random oracle model. The first AIBE scheme secure in the standard model was proposed by Gentry in [36] and the first AHIBE scheme also secure in the standard model was proposed by Boyen and Waters in [17]. Since then, there have been several other constructions with various useful properties and improvements [3, 4, 19, 23, 30, 45, 51, 53, 54, 57]. In Section 4.3, we provide a comparison of these constructions and a high-level intuition on how they work.

4.1 Setting

Definition 4.1: An anonymous identity-based encryption scheme, associated with an identity space $\mathcal{ISP}$, a message space $\mathcal{MSP}$, and a ciphertext space $\mathcal{CSP}$, is a tuple of PPT algorithms $(\mathsf{Setup}, \mathsf{Extract}, \mathsf{Encrypt}, \mathsf{Decrypt})$ such that:

$(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda)$: Setup takes the security parameter 1^λ as input and outputs the master public key MPK and the master secret key MSK.

$sk_I \leftarrow \mathsf{Extract}(\mathsf{MPK}, \mathsf{MSK}, I)$: $\mathsf{Extract}$ takes the master public key MPK, the master secret key MSK, and an identity $I \in \mathcal{ISP}$ as inputs and outputs a secret key sk_I for the identity I.

$c \leftarrow \mathsf{Encrypt}(\mathsf{MPK}, I, m)$: $\mathsf{Encrypt}$ takes the master public key MPK, an identity $I \in \mathcal{ISP}$, and a message $m \in \mathcal{MSP}$ as inputs and outputs a ciphertext $c \in \mathcal{CSP}$.

$m/\perp := \mathsf{Decrypt}(\mathsf{MPK}, sk_I, c)$: Given the master public key MPK, a secret key sk_I, and a ciphertext $c \in \mathcal{CSP}$, $\mathsf{Decrypt}$ either outputs a message $m \in \mathcal{MSP}$ or the failure symbol $\perp$. $\mathsf{Decrypt}$ is assumed to be deterministic.

Correctness. For every $I \in \mathcal{ISP}$, and every $m \in \mathcal{MSP}$, if sk_I is the secret key output by $\mathsf{Extract}(\mathsf{MPK}, \mathsf{MSK}, I)$ then $\mathsf{Decrypt}(\mathsf{MPK}, sk_I, \mathsf{Encrypt}(\mathsf{MPK}, I, m)) = m$. $\Diamond$

Definition 4.2: An anonymous hierarchical identity-based encryption scheme, associated with an identity space $\mathcal{ISP}$, a message space $\mathcal{MSP}$, and a ciphertext space $\mathcal{CSP}$, is a tuple of PPT algorithms $(\mathsf{Setup}, \mathsf{Extract}, \mathsf{Delegate}, \mathsf{Encrypt}, \mathsf{Decrypt})$ such that:

$\boldsymbol{(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, \ell)}$**:** Setup takes the security parameter 1^λ and the maximum depth of the hierarchy ℓ and outputs the master public key MPK and the master secret key MSK.

$\boldsymbol{sk_{\vec{I}} \leftarrow \mathsf{Extract}(\mathsf{MPK}, \mathsf{MSK}, \vec{I})}$**:** $\mathsf{Extract}$ takes the master public key MPK, the master secret key MSK, and an identity vector $\vec{I} \in \mathcal{ISP}^d$ such that $1 \le d \le \ell$ as inputs and outputs a secret key $sk_{\vec{I}}$ for the identity vector $\vec{I}$.

$\boldsymbol{sk_{\vec{I'}:I''} \leftarrow \mathsf{Delegate}(\mathsf{MPK}, sk_{\vec{I'}}, I'')}$**:** $\mathsf{Delegate}$ takes the master public key MPK, a secret key $sk_{\vec{I'}}$ for the identity vector $\vec{I'} \in \mathcal{ISP}^d$ such that $1 \le d < \ell$, and an identity $I'' \in \mathcal{ISP}$ as inputs and outputs a secret key $sk_{\vec{I'}:I''}$ for the identity vector $\vec{I'} : I'' \in \mathcal{ISP}^{d+1}$.

$\boldsymbol{c \leftarrow \mathsf{Encrypt}(\mathsf{MPK}, \vec{I}, m)}$**:** $\mathsf{Encrypt}$ takes the master public key MPK, an identity vector $\vec{I} \in \mathcal{ISP}^d$ such that $1 \le d \le \ell$, and a message $m \in \mathcal{MSP}$ as inputs and outputs a ciphertext $c \in \mathcal{CSP}$.

$\boldsymbol{m/\bot \coloneqq \mathsf{Decrypt}(\mathsf{MPK}, sk_{\vec{I}}, c)}$**:** Given the master public key MPK, a secret key $sk_{\vec{I}}$, and a ciphertext $c \in \mathcal{CSP}$, $\mathsf{Decrypt}$ either outputs a message $m \in \mathcal{MSP}$ or the failure symbol $\bot$. $\mathsf{Decrypt}$ is assumed to be deterministic.

Correctness. For every $\vec{I} \in \mathcal{ISP}^d$ such that $1 \le d \le \ell$, and every $m \in \mathcal{MSP}$, if $sk_{\vec{I}}$ is the secret key output by $\mathsf{Extract}(\mathsf{MPK}, \mathsf{MSK}, \vec{I})$ then $\mathsf{Decrypt}(\mathsf{MPK}, sk_{\vec{I}}, \mathsf{Encrypt}(\mathsf{MPK}, \vec{I}, m)) = m$. $\diamondsuit$

4.2 Security Models

In this section, we review the formal security models related to the anonymity notions in (hierarchical) identity-based encryption: AIBE and AHIBE. Following the work in the literature [3, 30, 45, 53], we present these security models as games played between a PPT challenger and an adversary. In a nutshell, the goal of the adversary in these games is to tell apart two ciphertexts generated under two different identities of which he does not own the corresponding secret keys. Depending on the game in question, the adversary is also granted some privileges.

We follow a unified approach in the presentation of these games. As such, we first present the games related to the X-CCA notions of security for X $\in$ {AIBE, AHIBE}. Next, we show how to tweak these games to obtain the X-CPA variations.

Definition 4.3: For a given A(H)IBE scheme, the X-IND-CCA game for X $\in$ {AIBE, AHIBE} played between a PPT challenger $\mathcal{C}$ and an adversary $\mathcal{A}$, is defined as:

Setup: $\mathcal{C}$ runs

$$(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \begin{cases} \mathsf{Setup}(1^\lambda) & \text{if X} = \text{AIBE} \\ \mathsf{Setup}(1^\lambda, \ell) & \text{otherwise} \end{cases}$$

and gives $\mathcal{A}$ the resulting master public key MPK, keeping the master secret key MSK to itself. $\mathcal{C}$ also initializes the set of revoked identities $R \coloneqq \emptyset$.

Phase 1: $\mathcal{A}$ adaptively issues queries $q_1, \ldots, q_m$ where each q_i is one of the following:

- Secret-key query $\vec{I}$: $\mathcal{A}$ requests the secret key of the identity $\vec{I} \in \mathcal{ISP}^d$ where $d = 1$ if X = AIBE, $1 \leq d \leq \ell$ otherwise. $\mathcal{C}$ runs $sk_{\vec{I}} \leftarrow \mathsf{Extract}(\mathsf{MPK}, \mathsf{MSK}, \vec{I})$, adds $\vec{I}$ to R, and sends $sk_{\vec{I}}$ to $\mathcal{A}$.
- Decryption query $(\vec{I}, c)$: $\mathcal{A}$ issues a decryption query on an identity $\vec{I} \in \mathcal{ISP}^d$, where $d = 1$ if X = AIBE, $1 \leq d \leq \ell$ otherwise, and a ciphertext $c \in \mathcal{CSP}$. $\mathcal{C}$ computes $sk_{\vec{I}} \leftarrow \mathsf{Extract}(\mathsf{MPK}, \mathsf{MSK}, \vec{I})$, runs $\mathsf{Decrypt}(\mathsf{MPK}, sk_{\vec{I}}, c)$, and gives the result to $\mathcal{A}$.

Challenge: $\mathcal{A}$ gives $\mathcal{C}$ two *equal* length messages $m_0^*, m_1^* \in \mathcal{MSP}$ and two identities $\vec{I}_0^*, \vec{I}_1^* \in \mathcal{ISP}^d$ with the following restrictions: 1. $(\mathsf{Prfx}(\vec{I}_0^*) \cup \mathsf{Prfx}(\vec{I}_1^*)) \cap R = \emptyset$. 2. $d = 1$ if X = AIBE, $1 \leq d \leq \ell$ otherwise. $\mathcal{C}$ picks $b^* \leftarrow\$ \{0,1\}$, computes $c^* \leftarrow \mathsf{Encrypt}(\mathsf{MPK}, \vec{I}_{b^*}^*, m_{b^*}^*)$, and sends c^* to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ adaptively issues additional queries $q_{m+1}, \ldots, q_n$ where each q_i is one of the following:

- Secret-key query $\vec{I}$ such that $\vec{I} \notin (\mathsf{Prfx}(\vec{I}_0^*) \cup \mathsf{Prfx}(\vec{I}_1^*))$.
- Decryption query $(\vec{I}, c)$ such that if $\vec{I} \in (\mathsf{Prfx}(\vec{I}_0^*) \cup \mathsf{Prfx}(\vec{I}_1^*))$, then $c \neq c^*$.

$\mathcal{C}$ replies in both cases as in *Phase 1*.

Guess: $\mathcal{A}$ outputs a guess $b \in \{0,1\}$ and wins if $b = b^*$.

The adversary $\mathcal{A}$ in this game is called an X-IND-CCA adversary and $\mathcal{A}$'s advantage is defined as

$$\mathsf{Adv}_{\mathcal{A},\Pi}^{\text{X-IND-CCA}} := \left|\Pr[b = b^*] - \tfrac{1}{2}\right|,$$

where the probability is over the random coins used by $\mathcal{C}$ and $\mathcal{A}$. ◊

Remark 4.1. In [18], Canetti et al. proposed a weaker notion of security called *selective-ID* security. In contrast to the *full* security game that we presented in Definition 4.3, the adversary in the selective-ID security game is required to output the challenge identities $\vec{I}_0^*, \vec{I}_1^*$ before the public parameters are generated by the challenger. As we will notice in Section 4.3, this weakened notion of security has allowed the realization of early AHIBE constructions.

Definition 4.4: An anonymous (hierarchical) identity-based encryption scheme Π is $(t, Q_{sk}, Q_d, \epsilon)$-X-CCA-secure for X $\in$ {AIBE, AHIBE} if for any t-time X-IND-CCA adversary making at most Q_{sk} (resp. Q_d) adaptive secret-key (resp. decryption) queries we have $\mathsf{Adv}_{\mathcal{A},\Pi}^{\text{X-IND-CCA}} \leq \epsilon$. ◊

By not allowing the adversary to submit any decryption queries in *Phase 1* and *Phase 2* of the X-IND-CCA game, one obtains the X-IND-CPA game.

Definition 4.5: An anonymous (hierarchical) identity-based encryption scheme Π is (t, Q_{sk}, ϵ)-X-CPA-secure for X $\in$ {AIBE, AHIBE} if Π is $(t, Q_{sk}, 0, \epsilon)$-X-CCA-secure. ◊

4.3 Constructions

In this section, we provide a high-level review of the existing A(H)IBE schemes in the literature. Table 2 shows a comparison of the existing AIBE schemes. Compared in Table 3 are the existing AHIBE schemes. As you can see in the case of AHIBE schemes, the depth parameters of the

Table 2: Comparison of the parameters of anonymous identity-based encryption schemes.

Scheme	ROM	Anonymity	Assumption
BF01 [12]	Yes	Full	BDH
Gen06 [36]	No	Full	q-ABDHE
AG09 [4]	Yes	Selective-ID	QR
AB09 [3]	No	Selective-ID	LWE in Lattices

Table 3: Comparison of the parameters of anonymous hierarchical identity-based encryption schemes. ℓ is the maximum depth of the hierarchy. d is the depth of the corresponding identity.

Scheme	MPK Length	sk Length	c Length	Anonymity	Decrypt Time	Group Order
BW06 [17]	$O(\ell^2)$	$O(\ell^2)$	$O(\ell)$	Selective-ID	$O(\ell)$	Prime
SW08 [57]	$O(\ell)$	$O((\ell-d)d)$	$O(\ell)$	Selective-ID	$O(d)$	Composite
SKOS09 [54]	$O(\ell)$	$O(\ell-d)$	$O(1)$	Selective-ID	$O(1)$	Composite
Duc10 [30]	$O(\ell)$	$O(\ell)$	$O(\ell)$	Selective-ID	$O(\ell)$	Prime
LL10 [45]	$O(\ell)$	$O(\ell-d)$	$O(1)$	Selective-ID	$O(1)$	Prime
DIP10 [23]	$O(\ell)$	$O(\ell-d)$	$O(1)$	Full	$O(1)$	Composite
SC11 [53]	$O(\ell)$	$O(\ell-d)$	$O(1)$	Full	$O(1)$	Composite
RWZ12 [51]	$O(\ell)$	$O(\ell-d)$	$O(1)$	Full	$O(1)$	Prime

hierarchy (ℓ and d) play a significant role in the efficiency of a given construction. Another crucial property increasing the efficiency of a scheme is basing the system on prime order groups.

The scheme of Boneh and Franklin [12] (BF01 in Table 2) was the first IBE scheme that was inherently anonymous. Their scheme was based on Weil pairing and its security was proven in the random oracle model. In 2006, Gentry [36] proposed the first AIBE scheme whose security was proven in the standard model (Gen06 in Table 2). His system was based on a new complexity assumption termed "decisional augmented bilinear Diffie-Hellman exponent (BDHE) assumption" which is related to the q-BDHE problem [10, 14]. In 2009, Ateniese and Gasti [4] proposed an AIBE scheme (AG09 in Table 2) that added anonymity to the non-anonymous IBE scheme of Cocks [21]. This construction was presented under the quadratic residuosity assumption and proven selective-ID secure in the random oracle model. In the same year, Agrawal and Boyen proposed another AIBE scheme based on hard problems on random integer lattices (AB09 in Table 2). The selective-ID security of this construction was proven in the standard model and its ciphertexts were pseudorandom.

Boyen and Waters presented the first AHIBE scheme provably secure without random oracles in their 2006 paper [17] (BW06 in Table 3). Their construction was based on the decision BDH and decision linear assumptions of prime order groups. A somewhat unattractive feature of this construction is the $O(\ell^2)$ length of the MPK and the user secret key. In [57], Shi and Waters constructed an AHIBE scheme (SW08 in Table 3) with $O(\ell)$ MPK length and $O((\ell-d)d)$ user secret key length. Their solution was based on several complexity assumptions of composite order groups. The first AHIBE scheme with constant length ciphertexts was proposed by Seo et al. in [54] (SKOS09 in Table 3). Their construction was also based on composite order groups. In [30], Ducas showed how to convert a large family of IBE and HIBE schemes into AIBE and AHIBE schemes by using asymmetric pairings. They also proposed a concrete AHIBE construction (Duc10 in Table 3) based on prime order groups. The first AHIBE scheme with constant ciphertext length *and* based on prime order groups was proposed by Lee and Lee in [45] (LL10 in Table 3). It is important to note that each of the AHIBE schemes mentioned above was proven secure in the selective-ID setting of anonymity.

The first AHIBE scheme achieving full security was proposed by De Caro et al. in [23]. This construction (DIP10 in Table 3) extended the HIBE scheme of Lewko and Waters [46] with anonymity properties. It should be noted that the scheme in [46] had already achieved full security and constant size ciphertexts. Seo and Cheon proposed another fully secure AHIBE scheme (SC11 in Table 3) by combining the HIBE scheme of [45] and the selective-ID secure AHIBE scheme of Seo et al. [54]. Finally, in [51], Ren et al. proposed the first fully secure AHIBE scheme (RWZ12 in Table 3) with constant ciphertext length based on prime order groups.

5 Anonymity in Broadcast Encryption

Broadcast Encryption (BE). Conventional encryption schemes as discussed in Section 3 and Section 4 provide the means for secret transmission of data in point-to-point communication. The setting of broadcast encryption, instead, consists of a *sender*, an insecure unidirectional *broadcast channel*, and a universe of *receivers*. When the sender wants to transmit some confidential data, it specifies the set of authorized receivers and creates an encrypted version of the content. A secure BE scheme enables legitimate receivers to recover the original content, while ensuring that excluded users just obtain meaningless data, even in the face of collusions.

The intrinsic access control capabilities of BE schemes make them a useful tool for many natural applications, spanning from protecting copyrighted content distributed as stored media [1], to managing digital subscriptions to satellite TV, to controlling access in encrypted file systems [15]. Thanks to its wide variety of applications, BE has received a lot of attention from the crypto research community in recent years (e.g., [14, 16, 25–28, 35, 39, 41, 50]). The quest in these works has been for ever more efficient solutions in terms of broadcast communication, key storage and encryption/decryption running time. And, in these respects, the constructions proposed in [14, 39] can be considered as being nearly optimal.

Unfortunately, in the setting of BE, little attention has been devoted to the exploration of refined security models that accurately account for the requirements inherent in multi-recipient communication. More specifically, the focus has been on providing assurance for sender-oriented properties, while overlooking the anonymity concerns of the receivers. This problem has also manifested in the definition of the decryption algorithm that explicitly requires the set of authorized receivers as an input. As a result, the state-of-the-art BE schemes are inherently incapable of preserving *any* notion of receiver anonymity.

Anonymous Broadcast Encryption (AnoBE). In certain broadcast applications, protecting the privacy of the receivers is just as important as preserving the confidentiality of the broadcast messages. For example, suppose a satellite TV provider employs a regular BE scheme to securely broadcast sensitive information over a channel to its subscribers. Now, since regular broadcast encryption does not provide any privacy guarantees of the users, a subscriber decrypting the channel using his secret key might also learn who else has subscribed for that channel. This is a serious violation of the privacy of the subscribers.

The first work in the literature to consider the recipient privacy/anonymity in broadcast encryption was put forth by Barth et al. [5]. Motivated by the privacy requirements in encrypted file systems, the authors therein introduced the notion of *private* broadcast encryption, aiming to prevent the leakage of the identities of the receivers. As a proof-of-concept, they also suggested two generic public-key constructions, one with decryption time linear in the number of legitimate recipients and another with constant decryption time, that do not leak any information about the set of authorized receivers and are secure in the standard model and in the random oracle model respectively. In [47], Libert et al. suggested a technique to prove the security of a variant of the second construction of [5] without reliance on random oracles, thus attaining an AnoBE construction with efficient decryption in the standard model. We review these AnoBE constructions in sufficient details in Section 5.3.

A major drawback of the AnoBE constructions mentioned above is that their ciphertexts have length linear in the number of authorized receivers. Furthermore, as Kiayias and Samari recently showed in [44], this drawback is unavoidable. In [44], the authors presented the lower bounds on the ciphertext length of AnoBE schemes and showed that fully anonymous broadcast encryption schemes with atomic ciphertexts (e.g., the schemes of [5, 47]) must have $\Omega(s \cdot \lambda)$ ciphertext length, where s is the number of authorized receivers and λ is the security parameter. This lower-bound highlights the cost of achieving full anonymity of the receivers in the setting of broadcast encryption.

Outsider-Anonymous Broadcast Encryption (oABE). There are yet other broadcast applications where the anonymity of the authorized receivers must be protected only from the outsiders (or the unauthorized users). As a simple example, imagine that a group of scientists working for a top secret government project wants to broadcast documents among themselves. Since their identities and the documents they share are equally sensitive, they decide to employ an AnoBE scheme for the transmissions. Now, since the scientists already know one another, the full anonymity provided by AnoBE is not really necessary. What they really need is a secure broadcast encryption scheme that prevents the leakage of their identities to the outsiders. Also, as shown in [44], full anonymity comes at a cost of ciphertext length being linear in the number of authorized receivers.

This notion of relaxed anonymity of the receivers was first formalized by Fazio and Perera in [33, 34] under the name outsider-anonymity. They identified that the notion of outsider-anonymity lies in between the complete lack of anonymity that characterizes traditional BE schemes, and the full anonymity provided by AnoBE. Taking advantage of this relaxation of anonymity, the authors in [33, 34] also presented generic oABE constructions with ciphertext length sub-linear in the number of legitimate receivers. We review these oABE constructions in Section 5.3 as well.

5.1 Setting

Definition 5.1: An (outsider-)anonymous broadcast encryption scheme, associated with a universe of users $U = [1, N]$, a message space $\mathcal{MSP}$, and a ciphertext space $\mathcal{CSP}$, is a tuple of PPT algorithms ($\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Encrypt}, \mathsf{Decrypt}$) such that:

$(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, N)$**:** Setup takes the security parameter 1^λ and the number of users in the system N and outputs the master public key MPK and the master secret key MSK.

$sk_i \leftarrow \mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i)$**:** KeyGen takes the master public key MPK, the master secret key MSK, and a user $i \in U$ as inputs and outputs a secret key sk_i for the user i.

$c \leftarrow \mathsf{Encrypt}(\mathsf{MPK}, S, m)$**:** $\mathsf{Encrypt}$ takes the master public key MPK, a set of receivers $S \subseteq U$, and a message $m \in \mathcal{MSP}$ as inputs and outputs a ciphertext $c \in \mathcal{CSP}$.

$\boldsymbol{m/\bot} :=$ **$\mathsf{Decrypt}(\mathsf{MPK}, sk_i, c)$:** Given the master public key MPK, a secret key sk_i, and a ciphertext $c \in \mathcal{CSP}$, $\mathsf{Decrypt}$ either outputs a message $m \in \mathcal{MSP}$ or the failure symbol $\bot$. $\mathsf{Decrypt}$ is assumed to be deterministic.

Correctness. For every $S \subseteq U$, every $i \in S$, and every $m \in \mathcal{MSP}$, if sk_i is the secret key output by $\mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i)$ then $\mathsf{Decrypt}(\mathsf{MPK}, sk_i, \mathsf{Encrypt}(\mathsf{MPK}, S, m)) = m$. $\Diamond$

Given above is the formal definition of an (outsider-)anonymous broadcast encryption as presented in [5,33,47]. Notice that the difference between this definition and the commonly accepted definition [39] of a regular BE scheme is that only in the latter does the decryption algorithm require the set of authorized users as an input. This is precisely why traditional BE scheme are inherently non-anonymous.

5.2 Security Models

In this section, we review the formal security models corresponding to the two notions of anonymity in broadcast encryption: AnoBE and oABE. First, we present the X-CCA notions of security for X $\in$ {AnoBE, oABE} as found in the literature [5,33,47]. These security notions are modeled as games, termed X-IND-CCA, played between a PPT challenger and an adversary. Next, we show how X-IND-CCA games can be tweaked to obtain the X-IND-CPA games corresponding to the X-CPA notions of security.

Definition 5.2: For a given AnoBE/oABE scheme, the X-IND-CCA game for X $\in$ {AnoBE, oABE} played between a PPT challenger $\mathcal{C}$ and an adversary $\mathcal{A}$, is defined as:

Setup: $\mathcal{C}$ runs $(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, N)$ and gives $\mathcal{A}$ the resulting master public key MPK, keeping the master secret key MSK to itself. $\mathcal{C}$ also initializes the set of revoked users $R := \emptyset$.

Phase 1: $\mathcal{A}$ adaptively issues queries $q_1, \ldots, q_m$ where each q_i is one of the following:

- Secret-key query i: $\mathcal{A}$ requests the secret key of a user $i \in U$. $\mathcal{C}$ runs $sk_i \leftarrow \mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i)$, adds i to R, and sends sk_i to $\mathcal{A}$.
- Decryption query (i, c): $\mathcal{A}$ issues a decryption query on a user $i \in U$ and a ciphertext $c \in \mathcal{CSP}$. $\mathcal{C}$ computes $sk_i \leftarrow \mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i)$, runs $\mathsf{Decrypt}(\mathsf{MPK}, sk_i, c)$, and gives the result to $\mathcal{A}$.

Challenge: $\mathcal{A}$ gives $\mathcal{C}$ two *equal* length messages $m_0^*, m_1^* \in \mathcal{MSP}$ and two equal length sets of user identities $S_0^*, S_1^* \subseteq U$ such that $\widehat{S} \cap R = \emptyset$ where

$$\widehat{S} = \begin{cases} S_0^* \triangle S_1^* & \text{if X = AnoBE} \\ S_0^* \cup S_1^* & \text{otherwise} \end{cases}.$$

$\mathcal{C}$ picks $b^* \leftarrow\$ \{0,1\}$, generates $c^* \leftarrow \mathsf{Encrypt}(\mathsf{MPK}, S_{b^*}^*, m_{b^*}^*)$, and sends c^* to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ adaptively issues additional queries $q_{m+1}, \ldots, q_n$ where each q_i is one of the following:

- Secret-key query i such that $i \notin \widehat{S}$.
- Decryption query (i, c) such that if $i \in \widehat{S}$, then $c \neq c^*$.

Table 4: Comparison of the parameters of (outsider-)anonymous broadcast encryption schemes. The second half shows the schemes with a tagging mechanism allowing only 1 decryption attempt per ciphertext. N is the total number of users in the system. r is the number of revoked users of a ciphertext.

Scheme	MPK Length	sk Length	c Length	Security	Anonymity	Decrypt Attempts
BBW06 [5]	$O(N)$	$O(1)$	$O(N-r)$	Static $\mathcal{A}$	Full	$(N-r)/2$
LPQ12 [47]	$O(N)$	$O(1)$	$O(N-r)$	Adaptive $\mathcal{A}$	Full	$(N-r)/2$
FP12a [33]	$O(1)$	$O(\log N)$	$O\Big(r\log\big(\frac{N}{r}\big)\Big)$	Adaptive $\mathcal{A}$	Outsider	$\Big(\big\lfloor r\log\big(\frac{N}{r}\big)\big\rfloor \log N\Big)/2$
BBW06* [5]	$O(N)$	$O(1)$	$O(N-r)$	Static $\mathcal{A}$, ROM	Full	1
LPQ12* [47]	$O(N)$	$O(1)$	$O(N-r)$	Adaptive $\mathcal{A}$	Full	1
FP12a* [33]	$O(N)$	$O(\log N)$	$O\Big(r\log\big(\frac{N}{r}\big)\Big)$	Adaptive $\mathcal{A}$	Outsider	1
FP12b [34]	$O(N\log N)$	$O(N)$	$O(r)$	Adaptive $\mathcal{A}$	Outsider	1

In both cases, $\mathcal{C}$ responds as in *Phase 1*.

Guess: $\mathcal{A}$ outputs a guess $b \in \{0,1\}$ and wins if $b = b^*$.

The adversary $\mathcal{A}$ in this game is called an X-IND-CCA adversary and $\mathcal{A}$'s advantage is defined as

$$\mathsf{Adv}^{\text{X-IND-CCA}}_{\mathcal{A},\Pi} := \left|\Pr[b = b^*] - \tfrac{1}{2}\right|,$$

where the probability is over the random coins used by $\mathcal{C}$ and $\mathcal{A}$. $\Diamond$

Definition 5.3: An (outsider-)anonymous broadcast encryption scheme Π is $(t, Q_{sk}, Q_d, \epsilon)$-X-CCA-secure for X $\in$ {AnoBE, oABE} if for any t-time X-IND-CCA adversary making at most Q_{sk} (resp. Q_d) adaptive secret-key (resp. decryption) queries we have $\mathsf{Adv}^{\text{X-IND-CCA}}_{\mathcal{A},\Pi} \leq \epsilon$. $\Diamond$

By restricting the adversary in the X-IND-CCA game from submitting any decryption queries during *Phase 1* and *Phase 2*, one obtains the X-IND-CPA game. The adversary is still allowed to issue secret-key queries.

Definition 5.4: An (outsider-)anonymous broadcast encryption scheme Π is (t, Q_{sk}, ϵ)-X-CPA-secure for X $\in$ {AnoBE, oABE} if Π is $(t, Q_{sk}, 0, \epsilon)$-X-CCA-secure. $\Diamond$

Notice that the security game in Definition 5.2 differs from the security game of the traditional BE schemes only in the *Challenge* step, where the BE adversary submits a single set of users while the AnoBE/oABE adversary submits two. The difference between the oABE security game and the AnoBE one lies in the domain of users the adversary is allowed to corrupt. Specifically, the oABE adversary is only allowed to corrupt the users not in $S_0^* \cup S_1^*$, whereas the AnoBE adversary is additionally allowed to corrupt the users in $S_0^* \cap S_1^*$ as well. This minor difference makes AnoBE constructions fully anonymous (also susceptible to the lower bound of linear ciphertext length [44]) and makes oABE constructions outsider anonymous.

5.3 Constructions

The prevailing approach of realizing AnoBE/oABE constructions has been *atomic* [44], i.e., their broadcast ciphertexts consist of multiple ciphertext components such that each component corresponds to a single recipient or a disjoint subset of recipients. To thwart CCA attacks, these ciphertext components are "sealed" together with a one-time signature scheme. A drawback of this approach is that a user decrypting a broadcast ciphertext has to *attempt* to decrypt each and every component until he hits a non-$\perp$ result. Therefore, in order to allow efficient identification of the correct ciphertext component, an anonymity-preserving tagging mechanism is sometimes employed.

A comparison of the parameters of the current AnoBE/oABE constructions is given in Table 4. The first half of this table shows the constructions without a tagging mechanism, and the second half show those with one thus requiring only one decryption attempt. We now provide a high-level review of the constructions given in this table.

AnoBE Constructions of [5]. After introducing a *static* notion of security for fully anonymous broadcast encryption (i.e., the adversary outputs the set of users he wants to corrupt before the setup step in the security game of Definition 5.2), Barth et al. also proposed two CCA-secure constructions of AnoBE in [5]. Their first construction (BBW06 in Table 4) employs an IK-CCA-secure public key encryption scheme Π' and a strongly existentially unforgeable one-time signature scheme Σ as underlying primitives. This construction works as follows.

$\mathsf{Setup}(1^\lambda, N)$: The Setup algorithm generates for each user $i \in U = [1, N]$ in the system a public/secret key pair (pk'_i, sk'_i) from Π' and sets $\mathsf{MPK} := \{pk'_i\}_{i \in U}$ and $\mathsf{MSK} := \{sk'_i\}_{i \in U}$.

$\mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i)$: A run of the KeyGen algorithm for a given user provides that user with the secret key sk'_i corresponding to his index $i \in U$.

$\mathsf{Encrypt}(\mathsf{MPK}, S, m)$: When encrypting a message m for a set of users $S \subseteq U$, $\mathsf{Encrypt}$ algorithm

1. generates a verification/signing key pair $(\mathsf{VK}, \mathsf{SK})$ of Σ,
2. for each user $j \in S$, generates a ciphertext component c_j by encrypting the message $\mathsf{VK} \| m$ under the public key pk'_i,
3. bundles the components $\{c_j\}_{j \in S}$ into $\widehat{c}$ after permuting them randomly,
4. generates the signature $\sigma \leftarrow \mathsf{Sign}(\mathsf{SK}, \widehat{c})$ and outputs $c = \sigma \| \widehat{c}$ as the broadcast ciphertext.

$\mathsf{Decrypt}(\mathsf{MPK}, sk_i, c)$: given a ciphertext c and a secret key sk'_i, the $\mathsf{Decrypt}$ algorithm

1. parses c as $\sigma \| \widehat{c}$, and $\widehat{c}$ as $c_1 \| \ldots \| c_s$,
2. using the given secret key sk'_i, attempts to decrypt each c_j for $j \in [1, s]$ until a non-$\perp$ result $\widehat{m}$ is obtained,
3. parses $\widehat{m}$ as $\mathsf{VK} \| m$ and checks whether $\mathsf{Vrfy}(\mathsf{VK}, \sigma, \widehat{c}) = \mathsf{True}$,
4. returns m if both step 2 and step 3 passes, otherwise returns $\perp$.

The security of BBW06 is proven by using a straightforward sequence-of-games argument. During each transition of the games the public key of a ciphertext component in the challenge ciphertext is switched from S_0^* to S_1^*. The indistinguishability of the transition is proven by reducing from the IK-CCA-security of the underlying public-key encryption scheme. A crucial step in the overall proof is the use of the unforgeability property of the underlying signature scheme, because of which the adversary is unable to extract a ciphertext component from the challenge ciphertext and create her own ciphertext to submit to the decryption oracle during *Phase 2*.

As you can see, the second step of the $\mathsf{Decrypt}$ algorithm above requires a receiver to *attempt* to decrypt $s/2$ components of the ciphertext on average. The purpose of the second construction of [5] (BBW06* in Table 4) is to bring down this number of decryption attempts to 1 while keeping the broadcast encryption scheme fully anonymous. The idea is to employ an anonymity-preserving labeling mechanism that directs the receivers to the correct ciphertext component.

In addition to the primitives used in BBW06, BBW06* uses a group $\mathbb{G} = \langle g \rangle$ with order q in which the CDH problem is assumed to be hard but the DDH problem is easy (i.e., there exists an efficient algorithm to test Diffie-Hellman tuples) and a hash function $H : \mathbb{G} \to \{0,1\}^\lambda$ which is modeled as a random oracle in the security analysis. BBW06* modifies BBW06 as follows.

- The Setup algorithm also generates for each user $i \in U$ an exponent $a_i \leftarrow\$ \mathbb{Z}_q$ and sets $A_i \coloneqq g^{a_i}$. Then, it adds $\{A\}_{i \in U}$ to MPK, and $\{a\}_{i \in U}$ to MSK.
- The KeyGen algorithm also provides the given user $i \in U$ with the secret exponent a_i.
- The $\mathsf{Encrypt}$ algorithm first generates the verification/signing key pair of Σ. Then, it picks $b \leftarrow\$ \mathbb{Z}_q$ and sets $B \coloneqq g^b$. Then, for each $j \in S$, it generates the ciphertext component c_j by encrypting $\mathsf{VK}\|A_j^b\|m$ under the public key pk'_j. Next, it attaches to each c_j the label $L_j = H(A_j^b)$. Finally, it attaches the group element B at the beginning of the randomly permuted ciphertext components and generates the signature σ on the entire ciphertext.
- The $\mathsf{Decrypt}$ algorithm first extracts from c the group element B. Next, it generates $L \coloneqq H(B^{a_i})$ and checks if $\exists_{j \in [1,s]}\ L = L_j$. If such j is found, it attempts to decrypt only the ciphertext component c_j using the secret key sk'_i. If that decryption results in a non-$\perp$ message $\widehat{m}$, in addition to verifying the signature using the verification key contained in $\widehat{m}$, it also checks if the group element contained in $\widehat{m}$ is indeed B^{a_i}. If any of the steps mentioned above fails, it outputs $\perp$. Otherwise, it outputs the message m contained in $\widehat{m}$.

Notice that the $\mathsf{Decrypt}$ algorithm still requires to do work on $s/2$ labels on average per ciphertext. However, since these operations are simple equality tests, their computational overhead is much smaller than multiple decryption attempts as in BBW06. As a consequence, we can safely say that BBW06* achieves constant decryption attempts.

The security of BBW06* is also proven using a sequence-of-games argument. Similar to BBW06, each transition switches the recipient of a ciphertext component of the challenge ciphertext from S_0^* to S_1^*, but it also involves 5 sub-transitions as shown below.

$$\begin{array}{lrl}
L_0: & \ldots H(g^{ba_{j_0}})\|\mathsf{Encrypt}'(pk'_{j_0}, \mathsf{VK}\|g^{ba_{j_0}}\|m_0)\ldots & \\
L_1: & \ldots H(g^{ba_{j_0}})\|\mathsf{Encrypt}'(pk'_{j_0}, R)\ldots & \text{for } R \leftarrow\$ \{0,1\}^{|\mathsf{VK}\|g^{ba_{j_0}}\|m_0|} \\
L_2: & \ldots \overline{R}\|\mathsf{Encrypt}'(pk'_{j_0}, R)\ldots & \text{for } \overline{R} \leftarrow\$ \{0,1\}^\lambda \\
L_3: & \ldots \overline{R}\|\mathsf{Encrypt}'(pk'_{j_1}, R)\ldots & \\
L_4: & \ldots H(g^{ba_{j_1}})\|\mathsf{Encrypt}'(pk'_{j_1}, R)\ldots & \\
L_5: & \ldots H(g^{ba_{j_1}})\|\mathsf{Encrypt}'(pk'_{j_1}, \mathsf{VK}\|g^{ba_{j_1}}\|m_1)\ldots &
\end{array}$$

The transitions from L_0 to L_1 and L_4 to L_5 are shown indistinguishable by using a simple reduction from the semantic security of the underlying public-key encryption Π'. The indistinguishability of the transition from L_2 to L_3 is shown by reducing from the IK-CCA-security of Π'. In order to prove that the transitions from L_1 to L_2 and L_3 to L_4 are indistinguishable, the authors model the hash function H as a random oracle with the intuition that the only way the adversary can tell the two games in each pair apart is by solving the CDH tuples (g, A_{j_0}, B) and (g, A_{j_1}, B) respectively.

AnoBE Constructions of [47]. Libert et al. revisited the notion of fully anonymous broadcast encryption in [47]. The main contributions of their paper are,

1. Propose an *adaptive* notion of security for AnoBE (i.e., Definition 5.2 when X = AnoBE)
2. Show that BBW06 construction is actually adaptive secure, rather than just statically secure as Barth et al. showed in [5].
3. Provide a construction (LPQ12 in Table 4) that is more efficient than BBW06 in terms of the constant factors hidden in the asymptotic ciphertext/secret key lengths.
4. Propose an anonymity-preserving labeling technique termed "anonymous hint system" that allows one to argue about the security of AnoBE in the standard model.
5. Provide an AnoBE construction (LPQ12* in Table 4) that incorporates an anonymous hint system to require only 1 decryption attempt per ciphertext and secure in the standard model.

The notion of anonymous hint system can be thought of as a generalization of the labeling mechanism needed to allow efficient decryption in an AnoBE scheme. Formally, an anonymous hint system associated with a tag space $\mathcal{TSP}$ consists of three algorithms as shown below.

$\boldsymbol{(pk, sk) \leftarrow}$ **KeyGen**$\boldsymbol{(1^\lambda)}$**:** KeyGen basically generates a public/secret key pair (pk, sk) given 1^λ.

$\boldsymbol{(U, H) \leftarrow}$ **Hint**$\boldsymbol{(t, pk, r)}$**:** Hint takes a tag $t \in \mathcal{TSP}$, a public key pk, and random coins $r \in \{0,1\}^\lambda$ and outputs a pair (U, H) with the requirement that U only depends on r and *not* on pk.

$\boldsymbol{H/\perp :=}$ **Invert**$\boldsymbol{(sk, t, U)}$**:** Invert basically attempts to invert the pair (t, U) given the secret key sk and obtain H. This algorithm is deterministic.

Libert et al. also formalizes two notions of security for an anonymous hint system: *anonymity* and *robustness*. Informally, anonymity requires that H does not leak any information about pk, whereas robustness prevents the creation of a tuple (U, H, t) such that H could be opened by inverting (U, t) under two distinct keys. Then, the authors also provide a concrete instantiation of an anonymous hint system borrowing the techniques from the twin DDH-based Cramer-Shoup cryptosystem of [20].

Finally, the LPQ12* construction, the most important contribution of [47], incorporates an anonymous hint system to a generic AnoBE-CCA-secure construction (such as BBW06 or LPQ12) to provide efficient decryption as follows. Without loss of generality we use the notation of BBW06 in the presentation of the LPQ12* construction below.

- The Setup algorithm also generates to each user $i \in U$ a private/secret key pair $(\overline{pk}_i, \overline{sk}_i)$ of the anonymous hint system. Then, it adds $\{\overline{pk}_i\}_{i\in U}$ to MPK and $\{\overline{sk}_i\}_{i\in U}$ to MSK.
- KeyGen also provides the given user $i \in U$ with the secret key $\overline{sk}_i$.
- After generating the signing/verification key pair $(\mathsf{VK}, \mathsf{SK})$ of Σ, the $\mathsf{Encrypt}$ algorithm picks $r \leftarrow\$ \{0,1\}^\lambda$ and computes $(U, H_j) \leftarrow \mathsf{Hint}(\mathsf{VK}, \overline{pk}_j, r)$ for each $j \in S$. Then, for each $j \in S$, it also generates the ciphertext component $c_j \leftarrow \mathsf{Encrypt}'(pk'_j, \mathsf{VK} \| m)$. Next, it attaches each

H_j to c_j, randomly permutes the resulting components, attaches $\mathsf{VK}\|U$ at the front of the permuted components, and obtains $\widehat{c}$. Finally, it generates $\sigma \leftarrow \mathsf{Sign}(\mathsf{SK}, \widehat{c})$ and outputs $c = \sigma\|\widehat{c}$ as the broadcast ciphertext.

- After verifying the signature, Decrypt algorithm extracts U from c, computes the hint $H := \mathsf{Invert}(\overline{sk}_i, \mathsf{VK}, U)$, and checks if $\exists_{j\in[1,s]}\ H = H_j$. If such j exists, it attempts to decrypt the corresponding c_j and obtains $\widehat{m}$. Then, it checks if the verification key contained in $\widehat{m}$ is indeed VK, the one attached to c in the clear. If any of the steps mentioned above fails, it outputs $\perp$. Otherwise, it outputs the message m contained in $\widehat{m}$.

It should be noted that the above construction uses the verification key of the signature scheme VK as the tag to the anonymous hint system. This has allowed Libert et al. to prevent the adversary in their security proof from reusing the components of the challenge ciphertext related to the anonymous hint system. The intuition is that the only way an adversary can successfully reuse those components is by knowing the corresponding signing key of the challenge verification key.

oABE Constructions of [33, 34]. Earlier we mentioned that Fazio and Perera proposed the notion of outsider-anonymity in broadcast encryption in [33,34]. The authors also presented the following practical constructions of oABE.

1. An oABE-CPA-secure scheme
2. An oABE-CCA-secure scheme (FP12a in Table 4)
3. An oABE-CCA-secure scheme with efficient decryption but secure in the random oracle model
4. An oABE-CCA-secure scheme with efficient decryption and secure in the standard model (FP12a* in Table 4)
5. An oABE-CCA-secure scheme with efficient decryption and extremely short ciphertexts and also secure in the standard model. (FP12b in Table 4)

A crucial primitive underlying these construction is the public-key variant [25] of the Subset Cover Framework [50]. Please refer to Section 2.3 for a review of this framework including its public-key variant. We now present a high-level intuition of the constructions of Fazio and Perera.

The idea of the most basic construction of [33], the oABE-IND-CPA-secure one, is to instantiate the CS method of the public-key SC framework [25] using an AIBE-CPA-secure anonymous identity-based encryption scheme (see Definition 4.1). The details of its Setup, KeyGen, Encrypt, and Decrypt algorithms of this construction follow in a similar fashion to the public-key CS method of [25] but with two crucial differences: 1. The Encrypt algorithm now has to make sure the ciphertext has exactly $\lfloor r\log(\frac{N}{r})\rfloor$ components, where r denotes the number of revoked users, in order to thwart trivial attacks caused by the non-uniform cover-set sizes. 2. Due to anonymity of the AIBE ciphertext components, the Decrypt algorithm can no longer identify the correct component, and as a consequence, attempts to decrypt each and every component resulting in $\left(\lfloor r\log(\frac{N}{r})\rfloor \log N\right)/2$ AIBE decryption attempts on average. The proof of security of this construction follows from a sequence-of-games argument where each transition is shown to be indistinguishable by a reduction from the AIBE-security of the underlying identity-based encryption scheme.

The authors build their second oABE construction by incorporating a strongly existentially unforgeable signature scheme to their first construction to provide oABE-CCA security. The way they incorporate this signature scheme is similar to that of the BBW06 construction.

Their third construction combines an anonymity-preserving tagging mechanism, which is similar to the one used by BBW06*, with their second construction to achieve an efficient decryption running time. Unfortunately, the security proof of this construction is in the random oracle model due to the way the tagging mechanism works.

The fourth construction of Fazio and Perera, which is given in the extended version [34] of [33], incorporates a different tagging mechanism to their third construction to achieve efficient decryption time and proof in the standard model. Similar to the anonymous hint system of [47], this tagging mechanism is also derived from the twin DDH-based Cramer-Shoup cryptosystem of [20].

Finally, the authors also present the sketch of an oABE-CCA secure scheme based on the public-key SD method of [25]. This construction achieves an extremely short ciphertext length of $O(r)$. However, it also has a MPK length of $O(N^2)$ and a sk length of $O(N)$ that might be troublesome for constrained environments.

An attractive feature enjoyed by these oABE constructions is that their ciphertext length is sub-linear in the number of recipients. For applications where the identity of the recipients must be protected only from the outsiders, oABE allows the most efficient broadcast encryption constructions to date. In fact, the ciphertext lengths achieved by the construction of Fazio and Perera are even *more* attractive than the most optimal ciphertext length of traditional non-anonymous BE schemes where each ciphertext must be implicitly attached a description of the intended set of recipients that could require either $O(s \log N)$ or $O(r \log N)$ storage overhead depending on the implementation.

6 Summary & Conclusion

In a nutshell, an anonymous encryption scheme prevents the leakage of the identities of the recipients from the ciphertexts. In a world where threats to personal privacy are increasing at an alarming rate, the need for recipient anonymity has become indispensable in many mainstream application domains of public-key cryptography.

In this paper, we reviewed the notion of anonymity in three mainstream application domains of public-key cryptography: standard public-key encryption, identity-based encryption, and broadcast encryption. Special consideration was given to the area of broadcast encryption due to our recent work in that setting and also our special interest to do further research in that direction.

For each of the aforementioned areas, we formally examined the setting and the game-based definition of anonymity in that setting. We also compared the existing concrete constructions providing anonymity guarantees in each setting. In the case of broadcast encryption, we also provided a thorough review of the inner workings of the currently available anonymous constructions. We hope the review presented in this paper will help find open problems in anonymous cryptographic systems to work on in the future.

References

[1] AACS. Advanced access content system. http://www.aacsla.com/.

[2] M. Abdalla, M. Bellare, D. Catalano, E. Kiltz, T. Kohno, T. Lange, J. Malone-Lee, G. Neven, P. Paillier, and H. Shi. Searchable encryption revisited: Consistency properties, relation to Anonymous IBE, and extensions. In *Advances in Cryptology—CRYPTO*, pages 205–222, 2005.

[3] S. Agrawal and X. Boyen. Identity-based encryption from lattices in the standard model. Manuscript, 2009. http://www.cs.stanford.edu/~xb/ab09/.

[4] G. Ateniese and P. Gasti. Universally anonymous ibe based on the quadratic residuosity assumption. In *Topics in Cryptology—CT-RSA*, pages 32–47, 2009.

[5] A. Barth, D. Boneh, and B. Waters. Privacy in encrypted content distribution using private broadcast encryption. In *Financial Cryptography and Data Security—FC*, pages 52–64, 2006.

[6] M. Bellare, A. Boldyreva, A. Desai, and D. Pointcheval. Key-privacy in public-key encryption. In *Advances in Cryptology—ASIACRYPT*, pages 566–582, 2001.

[7] M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway. Relations among notions of security for public-key encryption schemes. In *Advances in Cryptology—CRYPTO*, pages 26–45, 1998.

[8] M. Bellare and P. Rogaway. Optimal asymmetric encryption. In *Advances in Cryptology—EUROCRYPT*, pages 92–111, 1995.

[9] D. Boneh and X. Boyen. Secure identity based encryption without random oracles. In *Advances in Cryptology—CRYPTO*, pages 443–459, 2004.

[10] D. Boneh, X. Boyen, and E.-J. Goh. Hierarchical identity based encryption with constant size ciphertext. In *Advances in Cryptology—EUROCRYPT*, pages 440–456, 2005.

[11] D. Boneh, G. Crescenzo, R. Ostrovsky, and G. Persiano. Public key encryption with keyword search. In *Advances in Cryptology—EUROCRYPT*, pages 506–522, 2004.

[12] D. Boneh and M. K. Franklin. Identity-based encryption from the weil pairing. In *Advances in Cryptology—CRYPTO*, pages 213–229, 2001.

[13] D. Boneh, C. Gentry, and M. Hamburg. Space-efficient identity based encryption without pairings. In *IEEE Symposium on Foundations of Computer Science—FOCS*, pages 647–657, 2007.

[14] D. Boneh, C. Gentry, and B. Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys. In *Advances in Cryptology—CRYPTO*, pages 258–275, 2005.

[15] D. Boneh, E.-J. Goh, N. Modadugu, and H. Shacham. Sirius: Securing remote untrusted storage. In *ISOC Network and Distributed System Security Symposium—NDSS*, pages 131–145, 2003.

[16] D. Boneh and B. Waters. A fully collusion resistant broadcast, trace, and revoke system. In *ACM Conference on Computer and Communications Security—CCS*, pages 211–220, 2006.

[17] X. Boyen and B. Waters. Anonymous hierarchical identity-based encryption (without random oracles). In *Advances in Cryptology—CRYPTO*, pages 290–307, 2006.

[18] R. Canetti, S. Halevi, and J. Katz. A forward-secure public-key encryption scheme. *Journal of Cryptology*, 20(3):265–294, 2007.

[19] D. Cash, D. Hofheinz, E. Kiltz, and C. Peikert. Bonsai trees, or how to delegate a lattice basis. *Journal of Cryptology*, 25(4):601–639, 2012.

[20] D. Cash, E. Kiltz, and V. Shoup. The twin Diffie-Hellman problem and applications. In *Advances in Cryptology—EUROCRYPT*, pages 127–145, 2008.

[21] C. Cocks. An identity based encryption scheme based on quadratic residues. In *Cryptography and Coding*, pages 360–363, 2001.

[22] R. Cramer and V. Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In *Advances in Cryptology—CRYPTO*, pages 13–25, 1998.

[23] A. De Caro, V. Iovino, and G. Persiano. Fully secure anonymous hibe and secret-key anonymous ibe with short ciphertexts. In *Pairing-Based Cryptography—Pairing*, pages 347–366, 2010.

[24] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, Sept. 1976.

[25] Y. Dodis and N. Fazio. Public-key broadcast encryption for stateless receivers. In *Digital Rights Management—DRM*, pages 61–80, 2002.

[26] Y. Dodis and N. Fazio. Public-key trace and revoke scheme secure against adaptive chosen ciphertext attack. In *Public Key Cryptography—PKC*, pages 100–115, 2003.

[27] Y. Dodis, N. Fazio, A. Kiayias, and M. Yung. Scalable public-key tracing and revoking. In *ACM Symposium on Principles of Distributed Computing—PODC*, pages 190–199, 2003. Invited to the Special Issue of Journal of Distributed Computing PODC 2003.

[28] Y. Dodis, N. Fazio, A. Lysyanskaya, and D. Yao. ID-based encryption for complex hierarchies with applications to forward security and broadcast encryption. In *ACM Conference on Computer and Communications Security—CCS*, pages 354–363, 2004.

[29] D. Dolev, C. Dwork, and M. Naor. Non-malleable cryptography. *SIAM Journal on Computing*, 30(2):391–437, Apr. 2000.

[30] L. Ducas. Anonymity from asymmetry: New constructions for anonymous hibe. In *Topics in Cryptology—CT-RSA*, pages 148–164, 2010.

[31] M. Edman and B. Yener. On anonymity in an electronic society: A survey of anonymous communication systems. *ACM Computing Surveys*, 42(1):5:1–5:35, December 2009.

[32] T. Elgamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31(4):469–472, 1985.

[33] N. Fazio and I. M. Perera. Outsider-anonymous broadcast encryption with sublinear ciphertexts. In *Public Key Cryptography—PKC*, pages 225–242, 2012.

[34] N. Fazio and I. M. Perera. Outsider-anonymous broadcast encryption with sublinear ciphertexts. Cryptology ePrint Archive, Report 2012/129, 2012. Full Version of [33].

[35] J. A. Garay, J. Staddon, and A. Wool. Long-lived broadcast encryption. In *Advances in Cryptology—CRYPTO*, pages 333–352, 2000.

[36] C. Gentry. Practical identity-based encryption without random oracles. In *Advances in Cryptology—EUROCRYPT*, pages 445–464, 2006.

[37] C. Gentry and S. Halevi. Hierarchical identity based encryption with polynomially many levels. In *Theory of Cryptography—TCC*, pages 437–456, 2009.

[38] C. Gentry and A. Silverberg. Hierarchical ID-based cryptography. In *Advances in Cryptology—ASIACRYPT*, pages 548–566, 2002.

[39] C. Gentry and B. Waters. Adaptive security in broadcast encryption systems (with short ciphertexts). In *Advances in Cryptology—EUROCRYPT*, pages 171–188, 2009.

[40] S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984.

[41] D. Halevy and A. Shamir. The LSD broadcast encryption scheme. In *Advances in Cryptology—CRYPTO*, pages 47–60, 2002.

[42] J. Horwitz and B. Lynn. Toward hierarchical identity-based encryption. In *Advances in Cryptology—EUROCRYPT*, pages 466–481, 2002.

[43] J. Katz and Y. Lindell. *Introduction to Modern Cryptography.* Chapman & Hall/CRC, 2007.

[44] A. Kiayias and K. Samari. Lower bounds for private broadcast encryption. In *Information Hiding—IH*, pages 176–190, 2012.

[45] D. H. Lee and K. Lee. New techniques for anonymous hibe with short ciphertexts in prime order groups. *KSII Transactions on Internet and Information Systems (TIIS)*, 4(5):968–988, 2010.

[46] A. B. Lewko and B. Waters. New techniques for dual system encryption and fully secure hibe with short ciphertexts. In *Theory of Cryptography—TCC*, pages 455–479, 2010.

[47] B. Libert, K. G. Paterson, and E. A. Quaglia. Anonymous broadcast encryption. In *Public Key Cryptography—PKC*, pages 206–224, 2012.

[48] S. Luo, Y. Chen, J. Hu, and Z. Chen. New fully secure hierarchical identity-based encryption with constant size ciphertexts. In *Information Security Practice and Experience—ISPEC*, pages 55–70, 2011.

[49] R. C. Merkle. Secure communications over insecure channels. *Communications of the ACM*, 21(4):294–299, Apr. 1978.

[50] D. Naor, M. Naor, and J. Lotspiech. Revocation and tracing schemes for stateless receivers. In *Advances in Cryptology—CRYPTO*, pages 41–62, 2001.

[51] Y. Ren, S. Wang, and X. Zhang. Anonymous hierarchical identity-based encryption in prime order groups. In *Data and Knowledge Engineering—ICDKE*, pages 230–242, 2012.

[52] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, Feb. 1978.

[53] J. H. Seo and J. H. Cheon. Fully secure anonymous hierarchical identity-based encryption with constant size ciphertexts. Cryptology ePrint Archive, Report 2011/021, 2011.

[54] J. H. Seo, T. Kobayashi, M. Ohkubo, and K. Suzuki. Anonymous hierarchical identity-based encryption with constant size ciphertexts. In *Public Key Cryptography—PKC*, pages 215–234, 2009.

[55] A. Shamir. Identity-based cryptosystems and signature schemes. In *Advances in Cryptology—CRYPTO*, pages 47–53, 1984.

[56] C. E. Shannon. Communication theory of secrecy systems. *Bell System Technical Journal*, 28(4):656–715, 1949.

[57] E. Shi and B. Waters. Delegating capabilities in predicate encryption systems. In *Automata, Languages and Programming—ICALP*, pages 560–578, 2008.

[58] B. Waters. Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In *Advances in Cryptology—CRYPTO*, pages 619–636, 2009.