Generalized Identity Based and Broadcast Encryption Schemes*

Dan Boneh and Michael Hamburg

Stanford University
{dabo,mhamburg}@cs.stanford.edu

Abstract. We provide a general framework for constructing identity-based and broadcast encryption systems. In particular, we construct a general encryption system called *spatial encryption* from which many systems with a variety of properties follow. The ciphertext size in all these systems is independent of the number of users involved and is just three group elements. Private key size grows with the complexity of the system. One application of these results gives the first *broadcast HIBE* system with short ciphertexts. Broadcast HIBE solves a natural problem having to do with identity-based encrypted email.

1 Introduction

In this paper we develop a general framework for constructing identity-based encryption (IBE) [17,4] and broadcast encryption [9] with constant-size ciphertexts. This framework enables one to easily combine different encryption properties via a product rule and to obtain encryption systems supporting multiple properties. For example, a multi-authority, forward-secure, broadcast encryption system (with constant-size ciphertexts) is easily derived by taking the "product" of three systems. One new concept constructed using our framework is broadcast hierarchical IBE. We discuss this concept at the end of the section and explain its importance to secure email.

We start with an informal description of the framework; a precise definition is given in the next section. Rather than an IBE or a broadcast system we consider a higher level abstraction.

- Let $\mathcal{P}$ be a finite set of *policies*. Roughly speaking, a message m can be encrypted to any policy π in $\mathcal{P}$.
- Let $\mathcal{R}$ be a finite set of *roles*. Each decryptor has a role ρ in $\mathcal{R}$ and can obtain a private key K_ρ corresponding to its role ρ.
- We allow for an arbitrary predicate called open on the set $\mathcal{R}\times\mathcal{P}$ that specifies which roles in $\mathcal{R}$ can open what policies in $\mathcal{P}$.

A key K_ρ can decrypt ciphertexts encrypted for policy π if and only if role ρ opens policy π, i.e. $\mathsf{open}(\rho, \pi)$ is true.

To continue with the abstraction, we provide a notion of delegation which is useful in hierarchical IBE (HIBE) [13,11]. To support delegation we assume

* Supported by NSF and the Packard Foundation.

J. Pieprzyk (Ed.): ASIACRYPT 2008, LNCS 5350, pp. 455–470, 2008.

there is a partial order $\succeq$ defined on the set of roles $\mathcal{R}$. The idea is that given the key K_{ρ_1} there is a delegation algorithm that can be used to generate the key K_{ρ_2}, whenever $\rho_1 \succeq \rho_2$. Naturally, we require that the open relation respect delegation, meaning that if role ρ_2 opens policy π and $\rho_1 \succeq \rho_2$ then ρ_1 also opens π.

Given the sets $\mathcal{P}, \mathcal{R}$ and relations open and $\succeq$, one obtains a very general notion of identity-based encryption. It generalizes HIBE, broadcast encryption, attribute-based encryption [12], predicate encryption [6,14] and other variants. We refer to such schemes as generalized IBE, or GIBE. In the next section we define GIBE schemes more precisely along with their associated security games.

Spatial encryption. In Section 3 we study an important instance of GIBE called *spatial encryption* in which policies are points in $\mathbb{Z}_p^n$ and roles are affine subspaces of $\mathbb{Z}_p^n$. The delegation relation $\succeq$ on roles is defined by subspace inclusion: role $\rho_1 \succeq \rho_2$ if ρ_1's affine space contains ρ_2's space.

As we will see, spatial encryption enables us to build a host of identity-based and broadcast encryption schemes. In particular, it supports a product rule that lets us combine encryption properties such as forward security, multiple authorities, and others.

In Section 4 we construct an efficient spatial encryption system with constant-size ciphertext. Our starting point is an HIBE construction of Boneh, Boyen, and Goh [2]. We are able to extend their system to obtain a spatial encryption system. However, the proof of security is more difficult and requires the BDDHE assumption introduced in [5] (the proof in [2] used the slightly weaker BDHI assumption). We describe various extensions of the system at the end of Section 4.

Our initial motivation: email encryption. Suppose user A wishes to send an encrypted email to users $B_1, \ldots, B_n$. User A knows the identities of all recipients, but does not know which private key generators (PKGs) issued their private keys. Moreover, user A only trusts PKGs $P_1, \ldots, P_\ell$. She wishes to encrypt the email so that user B_j can decrypt it if and only if B_j has a private key issued by one of the ℓ trusted PKGs. Using basic IBE this will require ciphertext of size $O(n \cdot \ell)$. Our goal is to construct a system whose ciphertext size is constant, that is, independent of n and ℓ.

This natural email encryption problem can be modeled as a GIBE and constructed using the product of two instances of our spatial encryption scheme. Here each PKG has a role which can delegate to a key for any user; a (possibly distributed) dealer holds the master key $K_\top$. We obtain a system that precisely solves the problem described above, with ciphertext size independent of n and ℓ. However, in our current construction the private key size is linear in $n + \ell$.

Similarly, we also construct a broadcast HIBE. Roughly speaking, in a broadcast HIBE there is a tree-like hierarchy of identities and private keys as in HIBE. An encryptor picks a set S of nodes in the hierarchy and encrypts a message m to this set S. We let c be the resulting ciphertext. As in a broadcast system, any user in S can decrypt c, but (proper) coalitions outside of S cannot. We say that the system has constant-size ciphertext if the size of c is independent of the

size of S. Broadcast HIBE applies naturally to hierarchical email systems where messages can have many recipients.

Broadcast HIBE can be easily modeled as a GIBE and constructed from our spatial encryption system. This expands on the features of previous constant-size broadcast systems such as Boneh et al. [5] and Sakai and Furukawa [16], albeit at the cost of increased private-key size.

2 Generalized Identity-Based Encryption (GIBE)

A *Generalized Identity-Based Encryption Scheme*, or GIBE, allows a participant to encrypt a message under a certain *policy*, in some set $\mathcal{P}$ of allowable policies. We will enforce no structure on the allowed policies. To decrypt, users may hold secret keys corresponding to *roles*. Roles are organized in a partially-ordered set $\mathcal{R}$, that is, a set endowed with a reflexive, transitive, antisymmetric relation $\succeq$.

A GIBE may be parameterized in some way. For example, a system may have a limited number of identities, hierarchy levels, time periods or the like. We call such choices the *setup parameters* SP. As SP varies, $\mathcal{P}$ and $\mathcal{R}$ will generally also vary. Similarly, $\mathcal{P}$ and $\mathcal{R}$ may depend on the security parameter λ or on randomness chosen at setup. We encode these choices into a policy parameter χ generated at setup, and use policies $\mathcal{P}_\chi$ and roles $\mathcal{R}_\chi$. For brevity, we will omit χ when it is unambiguous.

For a policy π and a role ρ, we write $\mathsf{open}(\rho, \pi)$ if a user with a secret key for ρ is allowed to decrypt a message encrypted under π. We require this relation to be monotone, meaning that if $\rho \succeq \rho'$ and $\mathsf{open}(\rho', \pi)$ then $\mathsf{open}(\rho, \pi)$. For simplicity, we require that $\mathcal{R}$ contains a top element $\top$, such that $\top \succeq \rho$ for all $\rho \in \mathcal{R}$, and $\mathsf{open}(\top, \pi)$ for all $\pi \in \mathcal{P}$. Informally, greater roles open more messages, and the greatest role, $\top$, can open them all. Obviously, only a highly-trusted authority should hold the secret key $K_\top$.

A GIBE consists of four randomized algorithms:

- *Setup*(λ, SP) takes as input a security parameter λ and setup parameters SP. It returns public parameters PP (which include the policy parameter χ) and a master secret key $K_\top$.
- *Delegate*$(\mathrm{PP}, \rho, K_\rho, \rho')$ takes the secret key K_ρ for role ρ and returns a secret key $K_{\rho'}$ for ρ', where $\rho \succeq \rho'$.
- *Encrypt*(PP, π, m) encrypts a message m under a policy π.
- *Decrypt*$(\mathrm{PP}, \rho, K_\rho, \pi, c)$ decrypts a ciphertext c using a secret key K_ρ. Decryption may fail. However, we require that decryption succeeds when $\mathsf{open}(\rho, \pi)$, so that:

$$\textit{Decrypt}(\ \mathrm{PP},\ \rho,\ K_\rho,\ \pi,\ \textit{Encrypt}(\mathrm{PP},\ \pi,\ m)\ ) = m$$

  for all PP generated by *Setup*, for all policies π and roles ρ, and for all keys K_ρ for ρ delegated directly or indirectly from $K_\top$.

We require that the algorithms *Setup*, *Delegate*, *Encrypt*, *Decrypt* and the predicates open and $\succeq$ all run in expected polynomial time in λ. We also require that delegation is independent of the path taken; that is, if $\rho_1 \succeq \rho_2 \succeq \rho_3$, then

$$\mathit{Delegate}(\mathrm{PP}, \rho_1, K_{\rho_1}, \rho_3)$$

should produce the same distribution as

$$\mathit{Delegate}(\mathrm{PP}, \rho_2, \mathit{Delegate}(\mathrm{PP}, \rho_1, K_{\rho_1}, \rho_2), \rho_3)$$

2.1 Security

We define the security of a GIBE $\mathcal{I}$ in terms of a family of security games between a challenger and an adversary. The system parameters SP are fixed, and the adversary is allowed to depend on them. We define the full, CCA_2, anonymous game first (anonymity here refers to the property that the ciphertext leaks no information about the policy used to create it [1]).

Setup: The challenger runs $\mathit{Setup}(\lambda, \mathrm{SP})$ and sends PP to the adversary.

First query phase: The adversary makes several delegation queries ρ_i to the challenger, which runs $\mathit{Delegate}(\mathrm{PP}, \top, K_\top, \rho_i)$ and returns the resulting K_{ρ_i}. The adversary may also make decryption queries (ρ_i, π_i, c_i) to the challenger, where $\mathsf{open}(\rho_i, \pi_i)$. The challenger runs $K_{\rho_i} \leftarrow \mathit{Delegate}(\mathrm{PP}, \top, K_\top, \rho_i)$, then runs $\mathit{Decrypt}(\mathrm{PP}, \rho_i, K_{\rho_i}, \pi_i, c_i)$ and returns the resulting m_i (or fails).

Challenge: The adversary chooses messages m_0 and m_1 and policies π_0^* and π_1^*, and sends them to the challenger. We require that the adversary has not been given decryption keys for these policies, that is, $\neg\mathsf{open}(\rho_i, \pi_j^*)$ for all delegation queries ρ_i in the first query phase, and for $j \in \{0, 1\}$.

The challenger chooses a random $b \stackrel{\mathrm{R}}{\leftarrow} \{0, 1\}$, runs $\mathit{Encrypt}(\mathrm{PP}, \pi_b^*, m_b)$, and returns the resulting *challenge ciphertext* c^* to the adversary.

Second query phase: The second query phase is exactly like the first, except that the adversary may not issue decryption queries for c^*, and the adversary may not make delegation queries for roles that open π_j^* for $j \in \{0, 1\}$.

Guess: The adversary outputs a bit $b' \in \{0, 1\}$. The adversary wins if $b' = b$, and otherwise it loses.

There are several important variants on the above game:

- In a CCA_1 game, the adversary may not issue decryption queries during the second query phase.
- In a CPA game, the adversary may not issue decryption queries at all.
- In a non-anonymous game, we require that $\pi_0^* = \pi_1^*$.
- In a selective game, the setup phase is modified. The challenger sends the policy parameter χ to the adversary. The adversary chooses in advance its π_0^* and π_1^* and sends them to the challenger. Then the challenger sends the rest of the public parameters PP.

We define adversary $\mathcal{A}$'s advantage in game variant $\mathcal{V}$ (when $\mathcal{A}$ is attacking the GIBE system $\mathcal{I}$ with parameter SP) to be

$$\mathcal{V}\mathrm{Adv}_{\mathcal{A}\leftrightarrow(\mathcal{I},\mathrm{SP})}(\lambda) := |\ \Pr[\mathcal{A}\text{ wins }\mathcal{V}] - \Pr[\mathcal{A}\text{ loses }\mathcal{V}]\ |$$

We say that a GIBE $\mathcal{I}$ is $\mathcal{V}$-secure if for all setup parameters SP and all probabilistic polynomial-time adversaries $\mathcal{A}$, the function $\mathcal{V}\mathrm{Adv}_{\mathcal{A}\leftrightarrow(\mathcal{I},\mathrm{SP})}(\lambda)$ is a negligible function of λ.

In this paper we will primarily focus on the simplest security model, namely selective-security, non-anonymous, against a CPA adversary. We denote the adversary's advantage in this model by $(\mathrm{NonAnon}, \mathrm{Sel}, \mathrm{CPA})\mathrm{Adv}_{\mathcal{A}\leftrightarrow(\mathcal{I},\mathrm{SP})}(\lambda)$.

2.2 Example GIBE Instances

Many instances of GIBE already appear in the literature:

- In traditional IBE [17,4] the policies are simply identities and the roles are identities or $\top$. A message encrypted to an identity I can be decrypted only with a key for I or for $\top$. There is no delegation except from $\top$.
- In broadcast IBE [9] the policies are sets of identities and the roles are identities or $\top$. A message to a set S of identities can be decrypted only with a key for $I \in S$, or for $\top$. There is no delegation except from $\top$.
- In attribute-based encryption (ABE) [12], the policies are subsets of a set S of attributes, and the roles are upwardly closed subsets of $\top := 2^S$. A message to a set S of attributes can be decrypted with a key for any set containing S. [12] does not define a delegation model for attribute-based encryption, but the circuit-based implementation permits delegation by widening a k-of-n threshold gate into a $k+1$-of-$n+1$ threshold gate.
- In hierarchical IBE [13,11] the policies are identities and the roles are points in the hierarchy, with $\top$ at the root of the hierarchy. Here the key for a point x can either delegate to or decrypt from any point y below x.
- In forward-secure [7] systems, the roles and policies include a time t. Roles can be delegated by increasing the time t, and cannot decrypt messages with an earlier t.

The games used to define the security of these instances are special cases of the GIBE games. In the next section we will show that most of these instances can be constructed from a GIBE we call spatial encryption. These generic constructions for IBE and HIBE are competitive with the best known hand-tailored constructions. For broadcast IBE and forward-secure IBE our generic construction has short ciphertexts, but the private key is longer than the best known tailor-made constructions [7,16,5].

2.3 Embedding Lemmas

It is clear that some GIBEs can be used to construct other GIBEs. For example, it is obvious that any broadcast IBE can also function as a traditional IBE.

In particular, suppose that we have a GIBE $\mathcal{I}$ with policies $\mathcal{P}_\chi$ and roles $\mathcal{R}_\chi$, and we wish to define a GIBE with policies $\mathcal{P}'_\chi$ and roles $\mathcal{R}'_\chi$. Suppose that we are given an efficient injective map $f_\mathcal{P} : \mathcal{P}'_\chi \to \mathcal{P}_\chi$ and an efficient embedding $f_\mathcal{R} : \mathcal{R}'_\chi \to \mathcal{R}_\chi$ which satisfy $\mathsf{open}(f_\mathcal{R}(\rho), f_\mathcal{P}(\pi)) \iff \mathsf{open}(\rho, \pi)$ and $f_\mathcal{R}(\top) = \top$. Then we can define a GIBE $\mathcal{I}'$ with policies $\mathcal{P}'_\chi$ and roles $\mathcal{R}'_\chi$ simply by applying all $f_\mathcal{P}$ to all policies and $f_\mathcal{R}$ to all roles.

Lemma 1 (Embedding Lemma). *Let $\mathcal{I}$ and $\mathcal{I}'$ be GIBEs as defined above. For any GIBE adversary $\mathcal{A}$ against $\mathcal{I}'$, there is a GIBE adversary $\mathcal{B}$ against $\mathcal{I}$, running in about the same time as $\mathcal{A}$, such that*

$$\mathcal{V}\mathrm{Adv}_{\mathcal{A}\leftrightarrow(\mathcal{I},SP)}(\lambda) = \mathcal{V}\mathrm{Adv}_{\mathcal{B}\leftrightarrow(\mathcal{I},SP)}(\lambda)$$

Similarly, we can sometimes use collision-resistant hashing to construct new GIBEs. Suppose we have a GIBE $\mathcal{I}$ in which policies and roles are lists of elements of some set $\mathcal{X}$, and in which open and $\succeq$ are decided in a monotone fashion by comparing certain elements for equality. Suppose also that we have an efficient collision-resistant hash $H : \mathcal{X}' \to \mathcal{X}$ on some other set $\mathcal{X}'$. Then we can define a GIBE $\mathcal{I}'$ which is identical to $\mathcal{I}$ except that its policies and roles are lists over $\mathcal{X}'$ instead of $\mathcal{X}$, and all operations apply H pointwise to the policies and roles.

Lemma 2 (Hashed Embedding Lemma). *Let $\mathcal{I}$ and $\mathcal{I}'$ be GIBEs as defined above. For any GIBE adversary $\mathcal{A}$ against $\mathcal{I}'$, there is a GIBE adversary $\mathcal{B}_1$ against $\mathcal{I}$ and a collision-resistance adversary $\mathcal{B}_2$ against H, each running in about the same time as $\mathcal{A}$, such that*

$$\mathcal{V}\mathrm{Adv}_{\mathcal{A}\leftrightarrow(\mathcal{I},SP)}(\lambda) \leq \mathcal{V}\mathrm{Adv}_{\mathcal{B}_1\leftrightarrow(\mathcal{I},SP)}(\lambda) + \mathrm{CRAdv}_{\mathcal{B}_2\leftrightarrow H}(\lambda)$$

The proofs of these lemmas are immediate and are omitted.

3 Spatial Encryption: An Important Instance of GIBE

The building block for systems in our paper will be *spatial encryption*, a new GIBE. In spatial encryption, the policies $\mathcal{P}$ are the points of an n-dimensional affine space $\mathbb{Z}_q^n$. The roles $\mathcal{R}$ are all subspaces W of $\mathbb{Z}_q^n$ ordered by inclusion, and $\mathsf{open}(W, \pi) \iff W \ni \pi$.

3.1 Systems Derived from Spatial Encryption

To demonstrate the power of spatial encryption, we show that many other GIBEs are embedded in it.

Hierarchical IBE. Hierarchical IBE is trivially embeddable in spatial encryption. Here the path components are elements of $\mathbb{Z}_q$, and the paths are limited to length at most n. This extends easily to hierarchical IBE where the path components are strings by using the Hashed Embedding Lemma. This is not the only embedding of hierarchical IBE in spatial encryption, however.

Inclusive IBE. In *inclusive IBE*, the policies are subsets of size at most n of a set of identities. The roles are also subsets of size at most n, where $\rho \succeq \rho' \iff \rho \subseteq \rho'$; that is, one can delegate by adding elements to a set. We say that $\mathsf{open}(\rho, \pi)$ iff $\rho \subseteq \pi$; that is, a message to a set can be decrypted with a key for any subset.

We can embed inclusive IBE in a spatial system of dimension $n + 1$. Here the identities are elements of $\mathbb{Z}_q$, but this extends to inclusive IBE with strings as identities using the Hashed Embedding Lemma. We encode a policy $\pi \subset \mathbb{Z}_q$ as the coefficients of the polynomial $\hat{\pi}(t) := \prod_{c \in \pi} x - c$; this polynomial has degree at most n and therefore has at most $n + 1$ coefficients. We encode a role $\rho \subset \mathbb{Z}_q$ as the vector subspace of coefficients of polynomials which are divisible by $\prod_{c \in \rho} x - c$.

Inclusive IBE seems almost as powerful as spatial encryption; nearly all the applications in this paper use inclusive IBE rather than using spatial encryption directly.

Inclusive IBE can be built using attribute-based encryption, but this construction is less efficient than spatial encryption. In particular, the ciphertext has size $O(n)$. Our construction gives constant size ciphertext.

Co-inclusive IBE. Co-inclusive IBE is the dual of inclusive IBE. Policies and roles (other than $\top$) are sets of at most n identities, where $r \succeq r' \iff r \supseteq r'$; that is, one can delegate by removing elements from a set. We say that $\mathsf{open}(\rho, \pi)$ iff $\rho \supseteq \pi$; that is, a message to a set can be decrypted with a key for any set which contains it.

We can embed co-inclusive IBE in a spatial system of dimension $2n$. For a role ρ, we assign the span of $\{\boldsymbol{v}_i : i \in \rho\}$, where $\boldsymbol{v}_i = (1, i, i^2, \ldots, i^{2n-1})$ is the Vandermonde vector for i. To encrypt to a policy π, we encrypt to $\boldsymbol{v}_\pi := \sum_{i \in \pi} \boldsymbol{v}_i$. It is clear that $\boldsymbol{v}_\pi$ is not contained in the subspace for any role $\rho' \not\supseteq \pi$, for then we would have expressed $\boldsymbol{v}_\pi$ as a sum of at most $2n$ linearly independent vectors in two different ways.

Co-inclusive IBE can be built using attribute-based encryption, but this construction is less efficient than spatial encryption. Once again, the ciphertext has size $O(n)$. Our construction gives constant size ciphertext.

Broadcast Hierarchical IBE. Broadcast HIBE (and therefore also vanilla broadcast IBE [16]) is embeddable in inclusive IBE. The role for a path `a/b/c/...` in the hierarchy is the set $\{\mathtt{a}, \mathtt{a/b}, \mathtt{a/b/c}, \ldots\}$. The policy for a set of nodes in the hierarchy is the union of their roles. The scheme can broadcast to a set of points S in the hierarchy if the number of distinct path prefixes in S is less than the dimension n.

For a useful broadcast system, short ciphertexts are required. Our spatial encryption has constant-size ciphertexts, so our broadcast HIBE does as well.

Product Schemes. For GIBEs $\mathcal{I}_1, \mathcal{I}_2$ with roles $\mathcal{R}_1, \mathcal{R}_2$ and policies $\mathcal{P}_1, \mathcal{P}_2$, respectively, we define a product scheme $\mathcal{I}_1 \otimes \mathcal{I}_2$. This scheme's roles are $\mathcal{R}_1 \times \mathcal{R}_2$ and its policies are $\mathcal{P}_1 \times \mathcal{P}_2$. Here $\mathsf{open}((\rho_1, \rho_2), (\pi_1, \pi_2))$ if and only if $\mathsf{open}(\rho_1, \pi_1)$

and $\mathsf{open}(\rho_2, \pi_2)$, and similarly $(\rho_1, \rho_2) \succeq (\rho_1', \rho_2')$ if and only if $\rho_1 \succeq \rho_1'$ and $\rho_2 \succeq \rho_2'$. Note that this is different from what can be accomplished with double encryption, for here the recipient needs to be able to decrypt both components using a single key $K_{(\rho_1,\rho_2)}$. For instance, in the forward-secure encryption system that follows, a recipient decrypts with a key for a role ρ issued before time t, not a key for ρ and another key issued before time t.

Using the vector space $\mathbb{Z}_q^{n_1+n_2} \cong \mathbb{Z}_q^{n_1} \times \mathbb{Z}_q^{n_2}$, we can embed two instances of spatial encryption with dimensions n_1 and n_2 in one of dimension n_1+n_2. Therefore, if two schemes $\mathcal{I}_1$ and $\mathcal{I}_2$ are embeddable in spatial systems of dimensions n_1 and n_2, their product $\mathcal{I}_1 \otimes \mathcal{I}_2$ is embeddable in a spatial system of dimension $n_1 + n_2$. Similarly, we can construct product schemes in inclusive IBE. Here the policies are of the form $\pi_1 \uplus \pi_2$ and the roles are of the form $\rho_1 \uplus \rho_2$, where $\uplus$ denotes a disjoint union.

Multiple Authorities. A common limitation in IBE systems is the need to trust a single central authority. The central authority has the ability to decrypt any message sent using the system, but equally importantly, the central authority must correctly decide to whom it will issue keys for a given role. The human element of this authentication problem makes it less amenable to technical solutions.

Product schemes are a step toward a solution to this problem. Let $\mathcal{I}_a$ be a broadcast system whose identities are the names of authorities, and let $\mathcal{I}_s$ be any GIBE. Then the product system $\mathcal{I}_a \otimes \mathcal{I}_s$ is a multi-authority version of $\mathcal{I}_s$. A (possibly distributed) central dealer gives each authority a the decryption key for the role $(a, \top)$. Then if a user wishes to encrypt a message to some policy $\pi \in \mathcal{P}_s$, and trusts a set A of authorities, she encrypts the message to (A, π). This can be decrypted only by a user who holds the key for (a, ρ) where $a \in A$ and $\mathsf{open}(\rho, \pi)$, that is, one whom a has certified for a role which opens π.

Forward Security. There are already constructions of forward-secure IBE from HIBE, so we already know that forward-secure encryption is embeddable in spatial encryption [7]. We show a trivial forward-secure system from spatial encryption that will be useful in constructing product schemes. Set the policy for a time t to be the vector of t ones followed by $n - t$ zeros, and the role for a range of times $[t_1, t_2]$ to be the affine subspace of t_1 ones, followed by any $t_2 - t_1$ components, followed by $n - t_2$ zeros.

A similar construction works for forward-secure IBE based on inclusive IBE. These constructions require many more dimensions than [7], but they require the user to store only one secret key for a given range of times. This makes them more efficient for use in product schemes.

$\mathbf{CCA_2}$ Security. Following [3], we can use a MAC and a commitment scheme to create a CCA_2-secure encryption $\mathcal{I}'$ scheme from a scheme $\mathcal{I}$ which is merely CPA-secure. To encrypt a message m to a policy π, we choose a random MAC key k, a commitment com to k and the decommitment dec. We encrypt $c := \mathit{Encrypt}(\mathrm{PP}, (\pi, \mathrm{com}), (m, \mathrm{dec}))$ using the product $\mathcal{I}\otimes\mathrm{IBE}$, and set the ciphertext

as $(\mathrm{com}, c, \mathrm{MAC}(k, c))$. The resulting scheme is anonymous if $\mathcal{I}$ is, and fully secure if $\mathcal{I}$ is. The proof is exactly as in [3].

Email Encryption. We have now solved the motivating example of practical email encryption: by composing the above constructions, we can easily build a forward-secure, multiple-authority, CCA_2-secure broadcast hierarchical encryption system. This system can encrypt a message to n_r (path prefixes of) recipients, trusting in n_a authorities, with t time periods in a single key. The ciphertexts have constant size, and the private keys have size $O(n_a + n_r + t)$.

Short Identity-Based Ring Signatures. We can convert a GIBE $\mathcal{I}$ to an identity-based signature scheme using the product scheme $\mathcal{I} \otimes \mathrm{IBE}$. The signing key for a role ρ is $K_{(\rho,\top)}$, and a signature of a message m under a role ρ is $K_{(\rho,H(m))}$, where H is a collision-resistant hash. This construction has the curious property that a signature by ρ on a message m can be delegated to produce a signature by ρ' on m for any $\rho' \preceq \rho$. If this property is undesirable, delegation can be prevented by using $H((\rho, m))$ instead of $H(m)$ above.

If the construction of $\mathcal{I} \otimes \mathrm{IBE}$ is fully secure, then this signature scheme will be unforgeable; if it is selectively secure, then the signature scheme will be selectively unforgeable in the random oracle model for H.

If we choose $\mathcal{I}$ to be inclusive IBE, then this construction gives an identity-based ring signature system [15,8,18], in which a user A can sign messages anonymously on behalf of any set of users containing A. A straightforward implementation using spatial encryption would result in long signatures, but the length results from the ability to delegate signatures further. By removing this ability, we can build constant-length identity-based ring signatures. We give the details in the full version of the paper.

4 Constructing a Spatial Encryption System

We now turn to the construction of a selectively-secure n-dimensional spatial encryption system with constant-size ciphertext. Our construction is inspired by the construction of a constant size HIBE given in [2]. Our proof of security, however, requires a slightly stronger complexity assumption, namely the BDDHE assumption previously used in [5].

4.1 Notation

Vectors in this paper are always column vectors. When writing them inline, we transpose them to save space. We will be working with vectors of group elements, so we will adopt a convenient notation. For a vector $\boldsymbol{v} = (v_1, v_2, \ldots, v_n)^\top \in \mathbb{Z}_p^n$ of field elements, we use $g^{\boldsymbol{v}}$ to denote the vector of group elements

$$g^{\boldsymbol{v}} := (g^{v_1}, g^{v_2}, \ldots, g^{v_n})^\top \in \mathbb{G}^n$$

In many cases, we will manipulate these without knowing the actual vector $\boldsymbol{v}$. For example, given $g^{\boldsymbol{v}}$ and $\boldsymbol{w}$, we can easily compute $g^{\langle \boldsymbol{v}, \boldsymbol{w} \rangle}$, where $\langle \boldsymbol{v}, \boldsymbol{w} \rangle := \boldsymbol{v}^\top \boldsymbol{w}$ is the usual dot product on $\mathbb{Z}_p^n$.

We will write $\mathrm{Aff}(M, \boldsymbol{a}) \subseteq \mathbb{Z}_p^n$ for the d-dimensional affine space $\{M\boldsymbol{x} + \boldsymbol{a} : \boldsymbol{x} \in \mathbb{Z}_p^d\}$.

4.2 The System

The system parameters for our spatial encryption system will be a prime p (where $\log p$ is approximately the security parameter λ) and two groups $\mathbb{G}$ and $\mathbb{G}_T$ of order p, with a bilinear pairing $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$. Additionally, the public parameters will include group elements $g, g^{a_0}, t \in \mathbb{G}_T$ and a vector $g^{\boldsymbol{a}} \in \mathbb{G}^n$.

A secret key for an affine space $V := \mathrm{Aff}(M, \boldsymbol{x})$ will have the form

$$\left(g^r,\ g^{b + r a_0 + r\langle \boldsymbol{x}, \boldsymbol{a} \rangle},\ g^{r M^\top \boldsymbol{a}} \right)$$

where b is the master secret and r is random in $\mathbb{Z}_p$.

The four GIBE algorithms work as follows:

- *Setup*(λ, n) generates the system parameters $p, \mathbb{G}, \mathbb{G}_T$. It then chooses parameters
$$g \xleftarrow{\mathrm{R}} \mathbb{G}^*, \qquad a_0 \xleftarrow{\mathrm{R}} \mathbb{Z}_p, \qquad \boldsymbol{a} \xleftarrow{\mathrm{R}} \mathbb{Z}_p^n$$
and secret parameter $b \xleftarrow{\mathrm{R}} \mathbb{G}$, then computes $t := e(g, g)^b$. It outputs public parameters
$$\mathrm{PP} := (\ p,\ \mathbb{G},\ \mathbb{G}_T;\quad g,\ g^{a_0},\ g^{\boldsymbol{a}},\ t\ )$$
and master secret key
$$K_\top := \left(g,\ g^b,\ g^{\boldsymbol{a}} \right) \ \in \mathbb{G}^{n+2}$$
- *Delegate*$(\mathrm{PP}, V_1, K_{V_1}, V_2)$ takes two subspaces $V_1 := S(M_1, \boldsymbol{x}_1)$ and $V_2 := S(M_2, \boldsymbol{x}_2)$. Since V_2 is a subspace of V_1, we must have $M_2 = M_1 T$ and $\boldsymbol{x}_2' = \boldsymbol{x}_1 + M_1 \boldsymbol{y}$ for some (efficiently computable) matrix T and vector $\boldsymbol{y}$. We can then compute a key
$$\begin{aligned} \hat{K}_{V_2} &:= \left(g^r,\ g^{b + r a_0 + r\langle \boldsymbol{x}_1, \boldsymbol{a} \rangle} \cdot g^{r \boldsymbol{y}^\top M_1^\top \boldsymbol{a}},\ g^{r T^\top M_1^\top \boldsymbol{a}} \right) \\ &= \left(g^r,\ g^{b + r a_0 + r\langle \boldsymbol{x}_2, \boldsymbol{a} \rangle},\ g^{r M_2^\top \boldsymbol{a}} \right) \end{aligned}$$
for V_2. However, we also need to re-randomize it. To do this, we pick a random $s \xleftarrow{\mathrm{R}} \mathbb{Z}_p$ and compute
$$\begin{aligned} K_{V_2} &:= \left(g^r \cdot g^s,\ g^{b + r(a_0 + \langle \boldsymbol{x}_2, \boldsymbol{a} \rangle)} \cdot g^{s(a_0 + \langle \boldsymbol{x}_2, \boldsymbol{a} \rangle)},\ g^{r M_2^\top \boldsymbol{a}} \cdot g^{s M_2^\top \boldsymbol{a}} \right) \\ &= \left(g^{r+s},\ g^{b + (r+s)(a_0 + \langle \boldsymbol{x}_2, \boldsymbol{a} \rangle)},\ g^{(r+s) M_2^\top \boldsymbol{a}} \right) \end{aligned}$$
Notice that V_1 and V_2 may be the same subspace. In that case, this formula translates the secret key between different forms for V_1 and re-randomizes it. As a result, we are free to choose whatever representation of V we wish.

– *Encrypt*(PP, $\boldsymbol{x}, m$), where m is encoded as an element of the target group $\mathbb{G}_T$, picks a random $s \xleftarrow{R} \mathbb{Z}_p$ and computes a ciphertext

$$\left(g^s,\ g^{s(a_0+\langle \boldsymbol{x},\boldsymbol{a}\rangle)},\ m \cdot t^s \right)$$

– *Decrypt*(PP, $V, K_V, \boldsymbol{x}, c$) where $c = (c_1, c_2, c_3)$ is the above ciphertext, first delegates K_V to obtain the key $K_{\{\boldsymbol{x}\}} = (k_1, k_2) := \left(g^r, g^{b+r(a_0+\langle \boldsymbol{x},\boldsymbol{a}\rangle)} \right)$. It then recovers

$$\frac{c_3 \cdot e(c_2, k_1)}{e(c_1, k_2)} \ = \ \frac{m \cdot t^s \cdot e(g,g)^{rs(a_0+\langle \boldsymbol{x},\boldsymbol{a}\rangle}}{e(g,g)^{sb+rs(a_0+\langle \boldsymbol{x},\boldsymbol{a}\rangle)}} \ = \ m$$

4.3 Bilinear Decision Diffie-Hellman Exponent

To prove security we use a generalization of bilinear Diffie-Hellman first proposed in [5]. Let $\mathbb{G}$ be a group of prime order p, and let g be a generator of g. Let $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ be a bilinear map, and let n be a positive integer. We define the notation $g^{\alpha^{[a,b]}}$ for integers $a \leq b$ as

$$g^{\alpha^{[a,b]}} := \left(g^{\alpha^a}, g^{\alpha^{a+1}}, \ldots, g^{\alpha^b} \right)^\top$$

We then define distributions

$$\begin{aligned} \mathcal{P}_{\text{BDDHE}} := \text{choose: } & g \xleftarrow{R} \mathbb{G}^*, \alpha \xleftarrow{R} \mathbb{Z}_p, h \xleftarrow{R} \mathbb{G}^*, z \leftarrow e(g,h)^{\alpha^n} \\ \text{output: } & \left(g^{\alpha^{[0,n-1]}},\ g^{\alpha^{[n+1,2n]}},\ h,\ z \right) \end{aligned}$$

$$\begin{aligned} \mathcal{R}_{\text{BDDHE}} := \text{choose: } & g \xleftarrow{R} \mathbb{G}^*, \alpha \xleftarrow{R} \mathbb{Z}_p, h \xleftarrow{R} \mathbb{G}^*, z \xleftarrow{R} \mathbb{G}_T \\ \text{output: } & \left(g^{\alpha^{[0,n-1]}},\ g^{\alpha^{[n+1,2n]}},\ h,\ z \right) \end{aligned}$$

We define the BDDHE-advantage of a randomized algorithm $\mathcal{A} : \mathbb{G}^{2n+1} \times \mathbb{G}_T \to \{0,1\}$ as

$$\begin{aligned} \text{BDDHE Adv}_{\mathcal{A},n}(\lambda) := \Big| & \Pr\left[\mathcal{A}(x) = 1 : x \xleftarrow{R} \mathcal{P}_{\text{BDDHE}} \right] \\ & - \Pr\left[\mathcal{A}(x) = 1 : x \xleftarrow{R} \mathcal{R}_{\text{BDDHE}} \right] \Big| \end{aligned}$$

4.4 Proof of Selective Security

Call the spatial encryption system above $\mathcal{S}$. To make the proof more readable we abstract away re-randomization terms in the main proof of security. To do so, we divide the proof into two steps:

– First, we show in Observation 1 that if the system $\mathcal{S}$ is insecure then so is a system with rigged randomization parameters (i.e. a system where $a_0, \boldsymbol{a}, b, r$ and s are chosen non-uniformly). This step is straightforward.

– Second, we show in Theorem 1 that a specific rigging of the randomization parameters in $\mathcal{S}$ is secure. The combination of these two steps implies that $\mathcal{S}$ is secure.

We believe that hiding re-randomization terms in the main simulation makes the proof easier to understand.

Observation 1 (Rigged parameters). *Let $\mathcal{S}'$ be identical to $\mathcal{S}$ except that $a_0, \boldsymbol{a}, b$, the r in delegation queries and the s in the challenge ciphertext are chosen by some algorithm rather than uniformly at random. Then for any $\mathcal{V}$-adversary $\mathcal{A}$ against $\mathcal{S}$, there is is a $\mathcal{V}$-adversary $\mathcal{B}$ against $\mathcal{S}'$, running in about the same time as $\mathcal{A}$, such that*

$$\mathcal{V}\mathrm{Adv}_{\mathcal{A}\leftrightarrow(\mathcal{S},n)}(\lambda) = \mathcal{V}\mathrm{Adv}_{\mathcal{B}\leftrightarrow(\mathcal{S}',\backslash)}(\lambda)$$

Proof. The adversary $\mathcal{B}$ runs $\mathcal{A}$, but re-randomizes $\mathcal{A}$'s queries and the simulator's responses. More concretely, at setup time $\mathcal{B}$ chooses uniformly random $a_0' \xleftarrow{R} \mathbb{Z}_p, \boldsymbol{a}' \xleftarrow{R} \mathbb{Z}_p^n, b' \xleftarrow{R} \mathbb{Z}_p$. It sends $\mathcal{A}$ the public parameters

$$\left(p,\ \mathbb{G},\ \mathbb{G}_T;\quad g,\ g^{a_0+a_0'},\ g^{\boldsymbol{a}+\boldsymbol{a}'},\ t \cdot e(g,g)^{b'} \right)$$

$\mathcal{B}$ then adjusts $\mathcal{A}$'s queries to match these public parameters. For example, when $\mathcal{A}$ makes a delegation query, $\mathcal{B}$ passes the query through directly to the challenger. Given the response

$$\left(g^r,\ g^{b+ra_0+r\langle \boldsymbol{x},\boldsymbol{a}\rangle},\ g^{rM^\top \boldsymbol{a}} \right)$$

$\mathcal{B}$ computes a new key

$$\left(g^r,\ g^{b+ra_0+r\langle \boldsymbol{x},\boldsymbol{a}\rangle} \cdot g^{b'} \cdot (g^r)^{a_0'+\langle \boldsymbol{x},\boldsymbol{a}'\rangle},\ g^{rM^\top \boldsymbol{a}} \cdot (g^r)^{M^\top \boldsymbol{a}'} \right)$$

$\mathcal{B}$ re-randomizes it using *Delegate*, and returns it to $\mathcal{A}$.

Because $\mathcal{A}$'s view of the parameters is uniformly random, it is attacking the system $\mathcal{S}$. At the end, $\mathcal{B}$ will win its $\mathcal{S}'$-game if and only if $\mathcal{A}$ wins its $\mathcal{S}$-game, so

$$\mathcal{V}\mathrm{Adv}_{\mathcal{A}\leftrightarrow(\mathcal{S},n)}(\lambda) = \mathcal{V}\mathrm{Adv}_{\mathcal{B}\leftrightarrow(\mathcal{S}',\backslash)}(\lambda)$$

as claimed.

We now proceed to the selective-security game. Here we prove that spatial encryption is selectively CPA secure so long as the BDDHE-problem is hard on $\mathbb{G}$.

Theorem 1. *Let $\mathcal{A}$ be any non-anonymous, selective CPA adversary against $\mathcal{S}$. Then there is a BDDHE-adversary $\mathcal{B}$, running in about the same time as $\mathcal{A}$, such that:*

$$\mathrm{BDDHE}\,\mathrm{Adv}_{\mathcal{B},n+1}(\lambda) = \frac{1}{2} \cdot (\mathrm{NonAnon}, \mathrm{Sel}, \mathrm{CPA})\mathrm{Adv}_{\mathcal{A}\leftrightarrow(\mathcal{S},n)}(\lambda)$$

Proof. We first use the above observation to construct an $\mathcal{S}'$-adversary $\mathcal{A}'$ with the same advantage as $\mathcal{A}$. Our proof then follows by direct reduction. The simulator $\mathcal{B}$ takes $p, \mathbb{G}, \mathbb{G}_T$ and $(g^{\alpha^{[0,n]}},\ g^{\alpha^{[n+2,2n+2]}}, h, z)$ from the BDDHE problem above. For the setup phase, $\mathcal{B}$ passes to $\mathcal{A}'$ the policy parameters $\chi = (p, \mathbb{G}, \mathbb{G}_T, n)$. Upon receiving the intended target policy $\boldsymbol{v}$, the simulator sets

$$\boldsymbol{a} = \alpha^{[1,n]}, \qquad a_0 = -\langle \boldsymbol{v}, \boldsymbol{a} \rangle, \qquad b = \alpha^{n+1}$$

Note that while $\mathcal{B}$ cannot efficiently compute $\boldsymbol{a}, a_0$ or b, it can compute $g^{\boldsymbol{a}}, g^{a_0}$ and $e(g,g)^b$ which are all it needs to present the public parameters to $\mathcal{A}'$.

To answer delegation queries for a subspace $V = \mathrm{Aff}(M, \boldsymbol{x})$, the simulator finds a vector $\boldsymbol{u} = (u_1, u_2, \ldots, u_n)^\top$ such that $M^\top \boldsymbol{u} = 0$, but $\langle \boldsymbol{x} - \boldsymbol{v}, \boldsymbol{u} \rangle \neq 0$. Such a $\boldsymbol{u}$ must exist since $\boldsymbol{v} \notin V$, and it can easily be found by the Gram-Schmidt process. The simulator then formally sets

$$r = \frac{u_1 \alpha^n + u_2 \alpha^{n-1} + \ldots + u_n \alpha}{\langle \boldsymbol{x} - \boldsymbol{v}, \boldsymbol{u} \rangle}$$

Note that while $\mathcal{B}$ cannot efficiently compute r, it can compute g^r. Now, for any vector y, the coefficient of the missing term α^{n+1} in $r \langle \boldsymbol{y}, \boldsymbol{a} \rangle$ is exactly $\langle \boldsymbol{y}, \boldsymbol{u} \rangle / \langle \boldsymbol{x} - \boldsymbol{v}, \boldsymbol{u} \rangle$. Therefore, $rM^\top \boldsymbol{a}$ is a vector of polynomials in α of degree at most $2n$, and the coefficient of α^{n+1} is zero by the choice of $\boldsymbol{u}$. Therefore $\mathcal{B}$ can compute $g^{rM^\top \boldsymbol{a}}$ efficiently from $g^{\alpha^{[0,n]}}$ and $g^{\alpha^{[n+2,2n]}}$. Similarly, $\mathcal{B}$ can compute

$$\begin{aligned} g^{b + r(a_0 + \langle \boldsymbol{x}, \boldsymbol{a} \rangle)} &= g^{\alpha^n + r\langle \boldsymbol{v} - \boldsymbol{x}, \boldsymbol{a} \rangle} \\ &= g^{\alpha^n + P(\alpha) + \langle \boldsymbol{v} - \boldsymbol{x}, \boldsymbol{u} \rangle \alpha^n / \langle \boldsymbol{x} - \boldsymbol{v}, \boldsymbol{u} \rangle} \\ &= g^{P(\alpha)} \end{aligned}$$

where $P(\alpha)$ has degree $2n$ and a zero coefficient on the α^{n+1} term. $\mathcal{B}$ uses this technique to answer delegation queries during both query phases.

To construct a challenge ciphertext for the message m_i, the simulator formally sets $s = \log_g h$, returning $c = (h, z \cdot m)$.

$\mathcal{B}$ returns 1 if $\mathcal{A}'$ guesses correctly, and 0 otherwise. Now, if $z = e(g,h)^{\alpha^{n+1}}$, this is a valid challenge ciphertext, so $\mathcal{A}'$ wins with probability

$$\frac{1}{2} + \frac{1}{2} \cdot (\mathrm{NonAnon}, \mathrm{Sel}, \mathrm{CPA}) \mathrm{Adv}_{\mathcal{A}' \leftrightarrow (\mathcal{S}', \backslash)}(\lambda)$$

On the other hand, if z is random, then so is c and $\mathcal{A}'$ wins with probability $\frac{1}{2}$. As a result,

$$\begin{aligned} \mathrm{BDDHE}\ \mathrm{Adv}_{\mathcal{B}, n+1}(\lambda) &= \frac{1}{2} \cdot (\mathrm{NonAnon}, \mathrm{Sel}, \mathrm{CPA}) \mathrm{Adv}_{\mathcal{A}' \leftrightarrow (\mathcal{S}', \backslash)}(\lambda) \\ &= \frac{1}{2} \cdot (\mathrm{NonAnon}, \mathrm{Sel}, \mathrm{CPA}) \mathrm{Adv}_{\mathcal{A} \leftrightarrow (\mathcal{S}, n)}(\lambda) \end{aligned}$$

as claimed.

4.5 Extensions to Spatial Encryption

Short Public Parameters. The public parameters g, g^{a_0} and $g^{\boldsymbol{a}}$ in spatial encryption consist of uniformly random elements of $\mathbb{G}$ (with the caveat that $g \neq 1$). Therefore, given a random-oracle hash $H : [1, n+2] \to \mathbb{G}$, these parameters can be omitted.

Policy Delegation. It may be desirable to re-encrypt a message from a policy π to a more restrictive policy π'. A simple model of this is to make $\mathcal{P} \uplus \mathcal{R}$ into a partially-ordered set. We say that $\pi \succeq \pi'$ if π' can be delegated to π, and $\rho \succeq \pi$ if $\mathsf{open}(\rho, \pi)$. The bottom $\perp \in \mathcal{P}$ of the partially ordered set represents plaintext or plaintext-equivalent, i.e. a policy which anyone can decrypt. Then encryption becomes a special case of policy delegation, just as key generation is a special case of delegation.

We can implement policy delegation in spatial encryption by allowing encryptions to any affine subspace $W = \mathrm{Aff}(M, \boldsymbol{x}) \subset \mathbb{Z}_p^n$. This can be decrypted by a key K_V if and only if $V \cap W \neq \emptyset$. The encryptions look much like the private keys in Section 4.2:

$$\left(g^s,\ g^{s(a_0 + \langle \boldsymbol{x}, \boldsymbol{a} \rangle)},\ g^{sM^\top \boldsymbol{a}},\ m \cdot t^s \right)$$

This allows us to construct dual systems for many of the systems in Section 3, in which policies and roles are transposed. It also enables us to turn co-inclusive encryption into a k-of-n threshold system.

However, ciphertexts for the policy-delegated systems are no longer constant-size: their size is instead proportional to the dimension of the policy as a subspace of $\mathbb{Z}_p^n$. Furthermore, while the proof given in Section 4.4 still holds, the limitations of selective security seem much stronger: the adversary must choose a subspace to attack ahead of time.

5 Future Work

The biggest drawback of cryptosystems derived from spatial encryption is that our proof only shows selective security. We leave as a significant open problem the construction of a fully-secure spatial encryption system under a compact, refutable assumption (preferably one simpler than our BDDHE assumption). Since most of the systems derived in this paper can be constructed through inclusive IBE, a fully-secure inclusive system would be almost as strong a result. We note that Gentry's recent fully-secure "key-randomizable broadcast IBE" [10] is nearly identical to our inclusive IBE, except that Gentry's adversary is only allowed to issue delegation requests for singleton identities. This result suggests that a fully-secure inclusive IBE system is within reach.

Another important challenge is to reduce the the size of the secret keys. Our current construction requires users to store $O(n \log \lambda)$ bits of sensitive information in memory and on disk, which may be challenging in some scenarios.

6 Conclusions

We presented GIBE, a general framework for viewing identity-based and broadcast encryption systems. We also constructed a spatial encryption system, which is an important instance of GIBE. Spatial encryption supports a product rule which enables us to easily construct systems with various encryption properties. One result of spatial encryption is broadcast HIBE with short ciphertexts.

A natural open problem is to constuct a spatial encryption system where both ciphertexts and private keys are short. Perhaps the techniques in [5] or [16] can be used towards this goal.

Acknowledgement

Special thanks to Adam Barth for helpful discussions on multi-authority email encryption.

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. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 205–222. Springer, Heidelberg (2005)
2. Boneh, D., Boyen, X., Goh, E.: Hierarchical identity based encryption with constant size ciphertext. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 440–456. Springer, Heidelberg (2005)
3. Boneh, D., Canetti, R., Halevi, S., Katz, J.: Chosen-ciphertext security from identity-based encryption. SIAM J. of Computing (SICOMP) 36(5), 915–942 (2006)
4. Boneh, D., Franklin, M.: Identity-based encryption from the Weil pairing. SIAM Journal of Computing 32(3), 586–615 (2003); Extended abstract in Crypto 2001
5. Boneh, D., Gentry, C., Waters, B.: Collusion resistant broadcast encryption with short ciphertexts and private keys. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 258–275. Springer, Heidelberg (2005)
6. 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)
7. Canetti, R., Halevi, S., Katz, J.: A forward-secure public-key encryption scheme. Journal of Cryptology 20(3), 265–294 (2007); Early version in Eurocrypt 2003
8. Dodis, Y., Kiayias, A., Nicolosi, A., Shoup, V.: Anonymous identification in ad-hoc groups. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 609–626. Springer, Heidelberg (2004)
9. Fiat, A., Naor, M.: Broadcast encryption. In: Stinson, D.R. (ed.) CRYPTO 1993. LNCS, vol. 773, pp. 480–491. Springer, Heidelberg (1994)
10. Gentry, C.: Hierarchial identity based encryption with polynomially many levels. Personal communications (2008)
11. Gentry, C., Silverberg, A.: Hierarchical ID-based cryptography. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 548–566. Springer, Heidelberg (2002)

12. Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data. In: Proceedings of ACM CCS 2006 (2006)
13. Horwitz, J., Lynn, B.: Towards hierarchical identity-based encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 466–481. Springer, Heidelberg (2002)
14. 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)
15. Rivest, R., Shamir, A., Tauman, Y.: How to leak a secret. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 552–565. Springer, Heidelberg (2001)
16. Sakai, R., Furukawa, J.: Identity-based broadcast encryption (2007), http://eprint.iacr.org/2007/217
17. Shamir, A.: Identity-based cryptosystems and signature schemes. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (1985)
18. Zhang, F., Kim, K.: ID-based blind signature and ring signature from pairings. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 533–547. Springer, Heidelberg (2002)