Secure Identity Based Encryption Without Random Oracles

Dan Boneh[1]*, and Xavier Boyen[2]

[1] Computer Science Department, Stanford University, Stanford CA 94305-9045
dabo@cs.stanford.edu
[2] Voltage Security, Palo Alto, California
xb@boyen.org

Abstract. We present a fully secure Identity Based Encryption scheme whose proof of security does not rely on the random oracle heuristic. Security is based on the Decision Bilinear Diffie-Hellman assumption. This solves an open problem posed by Boneh and Franklin in 2001.

1 Introduction

Identity Based Encryption (IBE) provides a public key encryption mechanism where a public key is an arbitrary string such as an email address or a telephone number. The corresponding private key can only be generated by a Private Key Generator (PKG) who has knowledge of a master secret. In an IBE system, users authenticate themselves to the PKG and obtain private keys corresponding to their identities. Although Identity based encryption was proposed two decades ago [Sha84], and a few early precursors suggested over the years [Tan87,MY96], it is only recently that the first working implementations were proposed. Boneh and Franklin [BF01,BF03] defined a security model for Identity Based Encryption and gave a construction based on the bilinear Diffie-Hellman problem. Cocks [Coc01] describes another construction using quadratic residues modulo a composite. The security of these systems requires cryptographic hash functions that are modeled as random oracles, i.e., these systems are proven secure in the random oracle model [BR93]. The same holds for several other identity based systems featuring signatures [CC03], key exchange [SOK00], hierarchical identities [GS02], and signcryption [Boy03].

It is natural to ask whether secure IBE systems can exist in the standard model, i.e., without resorting to the random oracle heuristic. This question is especially relevant in light of several uninstantiable random oracle cryptosystems [CGH98,BBP04], which are secure in the random oracle model, but are trivially insecure under any instantiation of the oracle. Towards this goal, several recent results [CHK03,BB04,HK04] construct IBE systems secure without random oracles in weaker versions of the Boneh-Franklin model. However, until now, building a fully secure IBE remained open.

* Supported by NSF and the Packard Foundation.

In this paper we construct an IBE system that is secure in the Boneh-Franklin model without using random oracles. Security is based on the decisional version of the bilinear Diffie-Hellman assumption. Our system demonstrates that fully secure IBE systems can exist without random oracles. The main shortcoming of the proposed system is that it is inefficient; consequently, we mostly view our construction as an existence proof.

2 Preliminaries

Before presenting our results we briefly review a definition of security for an IBE system. We also review the definition for groups with a bilinear map. First, we introduce some notation.

2.1 Notation

For a finite set S we use $x \stackrel{\mathrm{R}}{\leftarrow} S$ to define a random variable x that picks an element of S uniformly at random. For a randomized algorithm $\mathcal{A}$ we use $x \stackrel{\mathrm{R}}{\leftarrow} \mathcal{A}(y)$ to define a random variable x that is the output of algorithm $\mathcal{A}$ on input y. We let $\Pr[b(x) : \ x \leftarrow \mathcal{A}(y)]$ denote the probability that the predicate $b(x)$ is true where x is the random variable defined by $x \leftarrow \mathcal{A}(y)$. For a vector $z \in \Sigma^n$ we use $z|_i$ to denote the i'th component of z.

2.2 Secure IBE Systems

Recall that an Identity Based Encryption system (IBE) consists of four algorithms [Sha84,BF01]: *Setup*, *KeyGen*, *Encrypt*, *Decrypt*. The *Setup* algorithm generates system parameters, denoted by *params*, and a master key *master-key*. The *KeyGen* algorithm uses the master key to generate the private key corresponding to a given identity. The encryption algorithm encrypts messages for a given identity (using the system parameters) and the decryption algorithm decrypts ciphertexts using the private key.

Boneh and Franklin [BF01] define chosen ciphertext security for IBE systems under a chosen identity attack. In their model the adversary is allowed to adaptively chose the public key it wishes to attack (the public key on which it will be challenged). More precisely, security for an IBE system is defined using the following two probabilistic experiments $\mathsf{CCA\text{-}Exp}_{\mathcal{A}}(0)$ and $\mathsf{CCA\text{-}Exp}_{\mathcal{A}}(1)$.

Experiment $\mathsf{CCA\text{-}Exp}_{\mathcal{A}}(b)$: for an algorithm $\mathcal{A}$ and a bit $b \in \{0,1\}$ define the following game between a challenger and $\mathcal{A}$:

Setup: A challenger runs the *Setup* algorithm. It gives $\mathcal{A}$ the resulting system parameters *params*. It keeps the corresponding *master-key* to itself.

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

– Private key query for an identity ID_i. The challenger responds by running algorithm *KeyGen* to generate the private key d_i corresponding to the public key ID_i. It sends d_i to $\mathcal{A}$.

– Decryption query for a ciphertext C_i and an identity ID_i. The challenger responds by running algorithm *KeyGen* to generate the private key d_i corresponding to ID_i. It then runs algorithm *Decrypt* to decrypt the ciphertext C_i using the private key d_i. It gives $\mathcal{A}$ the resulting plaintext.

These queries may be asked adaptively, that is, each query q_i may depend on the replies to $q_1, \ldots, q_{i-1}$.

Challenge: Once $\mathcal{A}$ decides that Phase 1 is over it outputs an identity ID^* and two equal length plaintexts $M_0, M_1 \in \mathcal{M}$ that it wishes to be challenged on, under the constraint that it had not previously asked for the private key of ID^*. The challenger sets the challenge ciphertext to $C^* = \mathit{Encrypt}(\mathit{params}, \mathsf{ID}^*, M_b)$. It sends C^* as the challenge to $\mathcal{A}$.

Phase 2: Algorithm $\mathcal{A}$ issues more queries $q_{m+1}, \ldots, q_n$ where q_i is one of:

- Private key query for any identity ID_i where $\mathsf{ID}_i \neq \mathsf{ID}^*$. The challenger responds as in Phase 1.
- Decryption query C_i for identity ID^* where $C_i \neq C^*$. The challenger responds as in Phase 1.

These queries may be asked adaptively as in Phase 1.

Guess: Finally, $\mathcal{A}$ outputs a guess $b' \in \{0, 1\}$.

We call b' the output of the game and define the random variable $\mathsf{CCA\text{-}Exp}_{\mathcal{A}}(b)$ as $\mathsf{CCA\text{-}Exp}_{\mathcal{A}}(b) = b'$. The probability is over the random bits used by the challenger and the adversary. We define adversary $\mathcal{A}$'s advantage in attacking the IBE system $\mathcal{E}$ as:

$$\mathrm{Adv}_{\mathcal{E},\mathcal{A}} = |\Pr[\mathsf{CCA\text{-}Exp}_{\mathcal{A}}(0) = 1] - \Pr[\mathsf{CCA\text{-}Exp}_{\mathcal{A}}(1) = 1]|.$$

Definition 1. *We say that an IBE system $\mathcal{E}$ is $(t, q_{\scriptscriptstyle ID}, q_{\scriptscriptstyle C}, \epsilon_{\scriptscriptstyle IBE})$-adaptive chosen ciphertext secure under a chosen identity attack if for any t-time* IND-ID-CCA *adversary $\mathcal{A}$ that makes at most $q_{\scriptscriptstyle ID}$ chosen private key queries and at most $q_{\scriptscriptstyle C}$ chosen decryption queries we have that $Adv_{\mathcal{E},\mathcal{A}} < \epsilon_{\scriptscriptstyle IBE}$. As shorthand, we say that $\mathcal{E}$ is $(t, q_{\scriptscriptstyle ID}, q_{\scriptscriptstyle C}, \epsilon_{\scriptscriptstyle IBE})$-*IND-ID-CCA *secure.*

Semantic Security. As usual, we define chosen plaintext security for an IBE system as in the game above, except that the adversary is not allowed to issue any decryption queries. The adversary may still issue adaptive private key queries. The resulting system is semantically secure under an adaptive chosen identity attack.

Definition 2. *We say that an IBE system $\mathcal{E}$ is $(t, q_{\scriptscriptstyle ID}, \epsilon_{\scriptscriptstyle IBE})$ chosen plaintext secure under a chosen identity attack if $\mathcal{E}$ is $(t, q_{\scriptscriptstyle ID}, 0, \epsilon_{\scriptscriptstyle IBE})$-chosen ciphertext secure under a chosen identity attack. As shorthand, we say that $\mathcal{E}$ is $(t, q_{\scriptscriptstyle ID}, \epsilon_{\scriptscriptstyle IBE})$-*IND-ID-CPA *secure.*

For $b \in \{0, 1\}$ we use $\mathsf{CPA\text{-}Exp}_{\mathcal{A}}(b)$ to denote the experiment $\mathsf{CCA\text{-}Exp}_{\mathcal{A}}(b)$ where $\mathcal{A}$ cannot make any decryption queries.

2.3 Bilinear Groups

We briefly review the necessary facts about bilinear maps and bilinear map groups.

1. $\mathbb{G}$ and $\mathbb{G}_1$ are two (multiplicative) cyclic groups of prime order p;
2. g is a generator of $\mathbb{G}$.
3. e is a bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_1$.

Let $\mathbb{G}$ and $\mathbb{G}_1$ be two groups as above. A bilinear map is a map $e : \mathbb{G}\times\mathbb{G} \to \mathbb{G}_1$ with the following properties:

1. Bilinear: for all $u, v \in \mathbb{G}$ and $a, b \in \mathbb{Z}$, $e(u^a, v^b) = e(u, v)^{ab}$.
2. Non-degenerate: $e(g, g) \neq 1$.

We say that $\mathbb{G}$ is a bilinear group if the group action in $\mathbb{G}$ can be computed efficiently and there exists a group $\mathbb{G}_1$ and an efficiently computable bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_1$ as above. Note that $e(\cdot, \cdot)$ is symmetric since $e(g^a, g^b) = e(g, g)^{ab} = e(g^b, g^a)$.

3 Complexity Assumptions

Let $\mathbb{G}$ be a bilinear group of prime order p and g be a generator of $\mathbb{G}$. We review the standard Bilinear Diffie-Hellman (BDH) assumption as well as the definition for binary biased Pseudo Random Functions (PRF's) and collision resistant functions.

3.1 Bilinear Diffie-Hellman Assumption

The BDH problem [Jou00,BF01] in $\mathbb{G}$ is as follows: given a tuple $g, g^a, g^b, g^c \in \mathbb{G}$ as input, output $e(g, g)^{abc} \in \mathbb{G}_1$. An algorithm $\mathcal{A}$ has advantage ϵ_{BDH} in solving BDH in $\mathbb{G}$ if

$$\Pr\left[\mathcal{A}(g, g^a, g^b, g^c) = e(g, g)^{abc}\right] \geq \epsilon_{\text{BDH}}$$

where the probability is over the random choice of a, b, c in $\mathbb{Z}_p$ and the random bits of $\mathcal{A}$.

Similarly, we say that an algorithm $\mathcal{B}$ that outputs $b \in \{0, 1\}$ has advantage ϵ_{BDH} in solving the *decision* BDH problem in $\mathbb{G}$ if

$$\left|\Pr\left[\mathcal{B}(g, g^a, g^b, g^c, e(g, g)^{abc}) = 0\right] - \Pr\left[\mathcal{B}(g, g^a, g^b, g^c, T) = 0\right]\right| \geq \epsilon_{\text{BDH}} \quad (1)$$

where the probability is over the random choice of a, b, c in $\mathbb{Z}_p$, the random choice of $T \in \mathbb{G}_1$ and the random bits of $\mathcal{B}$. We use the following notation:

- We denote the distribution over 5-tuples in the left term of (1) by $\mathcal{P}_{\text{BDH}}$.
- We denote the distribution over 5-tuples in the right term of (1) by $\mathcal{R}_{\text{BDH}}$.

Definition 3. *We say that the (t, ϵ_{BDH})-(Decision) BDH assumption holds in $\mathbb{G}$ if no t-time algorithm has advantage at least ϵ_{BDH} in solving the (decision) BDH problem in $\mathbb{G}$.*

Occasionally we drop the t and ϵ_{BDH} and refer to the BDH and Decision BDH assumptions in $\mathbb{G}$.

3.2 Biased Binary Pseudo Random Functions

Next we review the definition of a Pseudo Random Function (PRF) with bias δ. Let F be a function $F : \{0,1\}^w \to \{0,1\}$. We say that F has bias $\delta \in [0,1]$ if the expectation of F over all inputs in $\{0,1\}^w$ is δ, i.e., $(1/2^w)\sum_{x\in\{0,1\}^w} F(x) = \delta$.

We let Ω_δ denote the set of all functions $F : \{0,1\}^w \to \{0,1\}$ with bias δ. We also let K_1 denote a set of keys. For an algorithm $\mathcal{A}$ we define the following value:

$$\mathtt{Exp}_{\mathcal{A}}^{\Omega_\delta} = \Pr\left[\mathcal{A}^F(k_1) = 1 \ : \ F \overset{\mathrm{R}}{\leftarrow} \Omega_\delta,\ k_1 \overset{\mathrm{R}}{\leftarrow} K_1\right]$$

Here $\mathcal{A}^F(k_1)$ denotes the output of algorithm $\mathcal{A}$ when it is given oracle access to the function F and input k_1. The input k_1 is a dummy input needed only so that $\mathcal{A}$ takes the same input as the $\mathcal{A}$ below.

The biased Pseudo Random Functions that we will be using are parameterized by two random values, say $k_0 \in K_0$ and $k_1 \in K_1$. The parameter k_0 is kept secret while k_1 is public. To capture this concept we consider a set of functions $\mathcal{F} = \{F_{k_0,k_1} : \{0,1\}^w \to \{0,1\}\}_{k_0\in K_0, k_1\in K_1}$. For such a family of functions $\mathcal{F}$ and an algorithm $\mathcal{A}$ we define the following value:

$$\mathtt{Exp}_{\mathcal{A}}^{\mathcal{F}} = \Pr\left[\mathcal{A}^{F_{k_0,k_1}}(k_1) = 1 \ : \ k_0 \overset{\mathrm{R}}{\leftarrow} K_0,\ k_1 \overset{\mathrm{R}}{\leftarrow} K_1\right]$$

Note that $\mathcal{A}$ is given k_1 but is not given k_0.

Definition 4. *Let $\mathcal{F} = \{F_{k_0,k_1} : \{0,1\}^w \to \{0,1\}\}_{k_0\in K_0, k_1\in K_1}$ be a set of functions. We say that $\mathcal{F}$ is a $(\delta, t, \epsilon_{PRF}, q)$-biased-PRF if for any t-time oracle algorithm $\mathcal{A}$ making at most q queries to its oracle we have:*

$$\left|\mathit{Exp}_{\mathcal{A}}^{\Omega_\delta} - \mathit{Exp}_{\mathcal{A}}^{\mathcal{F}}\right| < \epsilon_{PRF}$$

We say that the parameter k_0 is kept secret while k_1 is public.

3.3 Collision Resistance

We briefly review the definition of collision resistant hash functions.

Definition 5. *Let Σ be an alphabet of size s and let n be some positive integer. We say that a family of functions $\mathcal{H} = \{H_k : \{0,1\}^w \to \Sigma^n\}_{k\in K}$ is (t, ϵ_H)-collision resistant if for any t-time algorithm $\mathcal{A}$ we have*

$$\Pr\left[H_k(x) = H_k(y) \ \mathit{and}\ x \neq y \ : \ k \overset{\mathrm{R}}{\leftarrow} K;\ (x,y) \overset{\mathrm{R}}{\leftarrow} \mathcal{A}(k)\right] < \epsilon_H$$

It is well known that collision resistant hash functions can be constructed from a finite cyclic group for which the discrete log problem is intractable. Since the Decision BDH assumption in $\mathbb{G}$ implies that discrete-log in $\mathbb{G}$ is intractable it follows that the existence of collision resistant hash functions is implied by the Decision BDH assumption. Consequently, rather than saying that our construction depends on both Decision BDH and collision-resistance we can say that our construction depends on Decision BDH alone for security. Nevertheless, in our security theorems we state collision resistance as an explicit assumption so that one can use any cryptographic hash function such as SHA-1, if so desired.

4 Secure IBE Construction

Before presenting our secure IBE system we first introduce a specific construction for a biased binary PRF from any collision resistant hash function. Later, in Section 5, we prove that it is indeed a PRF with overwhelming probability.

4.1 A Special Biased Binary PRF

Let Σ be an alphabet of size s, and let $\Sigma_\perp = \Sigma \cup \{\perp\}$. For $0 \leq m \leq n$, denote by $\Sigma^{(n,m)}$ the set of vectors in $\Sigma_\perp^n$ that have exactly m components in Σ. For any vector $K \in \Sigma^{(n,m)}$ with $n \geq m > 0$, and any function $H : \{0,1\}^w \to \Sigma^n$ with $w > 0$, we define the *bias map* $F_{K,H} : \{0,1\}^w \to \{0,1\}$ as

$$F_{K,H}(x) = \begin{cases} 0 & \text{if } \exists i \in \{1,\dots,n\} : H(x)|_i = K|_i \\ 1 & \text{if } \forall i \in \{1,\dots,n\} : H(x)|_i \neq K|_i \end{cases}$$

Observe that when H is a random function, the bias map $F_{K,H}$ has an expectation of $(1-1/s)^m$ over the inputs $x \in \{0,1\}^w$.

Definition 6. *Let n, m, w be positive integers with $m \leq n$. Let Σ be an alphabet of size s and set $\delta = (1-1/s)^m$. We say that a hash function family $\{H_k : \{0,1\}^w \to \Sigma^n\}_{k \in \mathcal{K}}$ is $(t, \epsilon_{PRF}, q, m)$-admissible if the function family $\{F_{K,H_k}\}_{K \in \Sigma^{(n,m)}, k \in \mathcal{K}}$ is a $(\delta, t, \epsilon_{PRF}, q)$-biased PRF. Here k is public and K is secret.*

In Section 5 we show how an admissible hash function family can be constructed given a collision resistant hash function family. In the rest of this section, we show how to use admissible hash functions to construct a secure IBE in the standard model.

4.2 Secure IBE Using Admissible Hash Functions

We are now ready to present our secure IBE system. It is based on a recent HIBE construction without random oracles by Boneh and Boyen [BB04] (secure in a selective identity attack model), itself inspired from a random oracle HIBE construction due to Gentry and Silverberg [GS02].

The system makes use of a collision resistant hash function and security is based on the Decision BDH assumption. Let $\mathbb{G}$ be a bilinear group of prime order p and g be a generator of $\mathbb{G}$. Let $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_1$ be the bilinear map. We assume that the messages to be encrypted are elements of $\mathbb{G}_1$.

Throughout the section we let $\Sigma = \{1, \dots, s\}$ be an alphabet of size s, although later we restrict our attention to the binary case $s = 2$. We also let $\{H_k : \{0,1\}^w \to \Sigma^n\}_{k \in \mathcal{K}}$ be a family of hash functions. For now, we assume that public keys (ID) are elements in $\{0,1\}^w$. We later extend the construction to public keys over $\{0,1\}^*$ by first hashing ID using a collision resistant hash $\tilde{H} : \{0,1\}^* \to \{0,1\}^w$. The IBE system works as follows:

***Setup*:** To generate system parameters the algorithm picks a random $\alpha \in \mathbb{Z}_p$ and sets $g_1 = g^\alpha$. Next, it picks a random $g_2 \in \mathbb{G}$ and a random $n \times s$ matrix $U = (u_{i,j})$ where each $u_{i,j}$ is random in $\mathbb{G}$. Finally, the algorithm picks a random $k \in \mathcal{K}$ as a hash function key. The system parameters are $params = (g, g_1, g_2, U, k)$. the master key is $master\text{-}key = g_2^\alpha$.

***KeyGen*(*params*, ID, *master-key*):** To generate the private key for an identity $\mathsf{ID} \in \{0,1\}^w$ the algorithm lets $\bar{a} = H_k(\mathsf{ID}) = a_1 \dots a_n \in \Sigma^n$ and picks random $r_1, \dots, r_n \in \mathbb{Z}_p$. The private key d_{ID} is:

$$d_{\mathsf{ID}} = \left(g_2^\alpha \cdot \prod_{i=1}^{n} u_{i,a_i}^{r_i}, \quad g^{r_1}, \quad \dots, \quad g^{r_n} \right) \in \mathbb{G}^{n+1}$$

***Encrypt*(*params*, ID, M):** To encrypt a message $M \in \mathbb{G}_1$ under the public key $\mathsf{ID} \in \{0,1\}^w$, first set $\bar{a} = H_k(\mathsf{ID}) = a_1 \dots a_n \in \Sigma^n$, then pick a random $t \in \mathbb{Z}_p$ and output

$$C = \left(e(g_1, g_2)^t \cdot M, \quad g^t, \quad u_{1,a_1}^t, \quad \dots, \quad u_{n,a_n}^t \right)$$

Note that $e(g_1, g_2)$ can be precomputed so that encryption does not require any pairing computations.

***Decrypt*(*params*, d_{ID}, C):** To decrypt a ciphertext $C = (A, B, C_1, \dots, C_n)$ using the private key $d_{\mathsf{ID}} = (d_0, d_1, \dots, d_n)$, output:

$$A \cdot \frac{\prod_{j=1}^{n} e(C_j, d_j)}{e(B, d_0)} = M$$

Let $\bar{a} = H_k(\mathsf{ID}) = a_1 \dots a_n \in \Sigma^n$. Then, indeed, for a valid ciphertext we have:

$$\frac{\prod_{j=1}^{n} e(C_j, d_j)}{e(B, d_0)} = \frac{\prod_{j=1}^{n} e(u_{j,a_j}, g)^{tr_j}}{e(g, g_2)^{t\alpha} \prod_{j=1}^{n} e(g, u_{j,a_j})^{tr_k}} = \frac{1}{e(g_1, g_2)^t}$$

This completes the description of the system.

4.3 Security

We now turn to proving security of the IBE above. The system makes use of an admissible hash function family and security is based on the Decision BDH assumption. We prove security in the standard model, i.e., without random oracles.

Theorem 1. *Let $|\Sigma| = s$. Suppose the (t, ϵ_{BDH})-Decision BDH assumption holds in $\mathbb{G}$. Furthermore, suppose $\{H_k : \{0,1\}^w \to \Sigma^n\}_{k \in \mathcal{K}}$ is a $(t, \epsilon_{PRF}, q+1, m)$-admissible family of hash functions. Set $\delta = (1 - 1/s)^m$ and $\Delta = \delta(1-\delta)^q$. Assume that $\Delta > \epsilon_{PRF}$. Then the IBE system above is (t, q, ϵ_{IBE})-chosen plaintext* (IND-ID-CPA) *secure for any $\epsilon_{IBE} \geq 2\epsilon_{BDH}/(\Delta - \epsilon_{PRF})$.*

We note that taking $m = \Theta(s \log q)$ leads to $\Delta = \Theta(1/q)$. Then, ignoring ϵ_{PRF}, we have that $\epsilon_{\mathrm{IBE}} = \Theta(q\epsilon_{\mathrm{BDH}})$. Hence, in groups where $(t, \epsilon_{\mathrm{BDH}})$-Decision BDH holds we obtain a $(t, q, \Theta(q\epsilon_{\mathrm{BDH}}))$ secure IBE system without random oracles.

To prove the theorem we need to show that for any t-time algorithm $\mathcal{A}$ that makes at most q private key queries we have

$$\left| \Pr\left[\mathsf{CPA\text{-}Exp}_{\mathcal{A}}(0) = 1\right] - \Pr\left[\mathsf{CPA\text{-}Exp}_{\mathcal{A}}(1) = 1\right] \right| < \epsilon_{\mathrm{IBE}}$$

To do so we first define two additional experiments.

Experiment 1: $\mathsf{BDH\text{-}Exp}_{\mathcal{A}}(b, (g, g_1, g_2, g_3, T))$. Let $\mathcal{A}$ be an algorithm, b be a bit in $\{0, 1\}$, and (g, g_1, g_2, g_3, T) a 5-tuple where $g, g_1, g_2, g_3 \in \mathbb{G}$ and $T \in \mathbb{G}_1$. Define the following game between a simulator and $\mathcal{A}$:

Setup: To start, the simulator generates system parameters by first picking a random vector $V = v_1 \dots v_n \in \Sigma^{(n,m)}$. It then generates an $n \times s$ matrix $U = (u_{i,j})$ as follows. For each $i = 1, \dots, n$ and $j = 1, \dots, s$ it picks a random $\alpha_{i,j} \in \mathbb{Z}_p$ and sets

$$u_{i,j} = \begin{cases} g_2 \cdot g^{\alpha_{i,j}} & \text{if } v_i = j, \text{ and} \\ g^{\alpha_{i,j}} & \text{otherwise} \end{cases}$$

Next, the simulator picks a random $k \in \mathcal{K}$ as a hash function key. It gives $\mathcal{A}$ the system parameters $params = (g, g_1, g_2, U, k)$. Note that the corresponding (unknown) master key is $master\text{-}key = g_2^{\alpha}$ where $\alpha = \log_g g_1$.

Phase 1. $\mathcal{A}$ issues up to q private key queries. Consider a query for the private key $\mathsf{ID} \in \{0, 1\}^w$. Let $\bar{a} = H_k(\mathsf{ID}) = a_1 \dots a_n \in \Sigma^n$. If $a_i \neq v_i$ for all $i = 1, \dots, n$ then the simulator terminates the experiment and outputs abort. Otherwise, there exists an i such that $a_i = v_i \in \Sigma$. The simulator derives the private key for ID by first picking random elements $r_1, \dots, r_n \in \mathbb{Z}_p$ and then setting

$$d_0 = g_1^{-\alpha_{i,v_i}} \prod_{j=1}^{n} u_{j,a_j}^{r_j}, \quad d_1 = g^{r_1}, \quad \dots, \quad d_i = g^{r_i}/g_1, \quad \dots, \quad d_n = g^{r_n} \quad (2)$$

We note that $(d_0, d_1, \dots, d_n) \in \mathbb{G}^{n+1}$ is a valid random private key for ID. To see this, let $\tilde{r}_i = r_i - \alpha$. Then we have that

$$g_1^{-\alpha_{i,v_i}} \prod_{j=1}^{n} u_{j,a_j}^{r_j} = g_2^{\alpha} \cdot (g_2\, g^{\alpha_{i,v_i}})^{-\alpha} \cdot \prod_{j=1}^{n} u_{j,a_j}^{r_j} = g_2^{\alpha} \cdot u_{i,a_i}^{\tilde{r}_i} \cdot \prod_{j=1, j\neq i}^{n} u_{j,a_j}^{r_j}$$

It follows that the key $(d_0, d_1, \dots, d_n)$ defined in (2) satisfies:

$$d_0 = g_2^{\alpha} \cdot (u_{i,a_i}^{\tilde{r}_i} \cdot \prod_{j=1, j\neq i}^{n} u_{j,a_j}^{r_i}), \quad d_1 = g^{r_1}, \quad \dots, \quad d_i = g^{\tilde{r}_i}, \quad \dots, \quad d_n = g^{r_n}$$

where $r_1, \dots, \tilde{r}_i, \dots, r_n$ are uniform in $\mathbb{Z}_p$. This matches the definition for a private key for ID and hence $(d_0, d_1, \dots, d_n)$ is a valid private key for ID. The simulator gives this key to $\mathcal{A}$.

Challenge. $\mathcal{A}$ outputs an identity ID^* and two messages $M_0, M_1 \in \mathbb{G}_1$. Let $\bar{a} = H_k(\mathsf{ID}^*) = a_1 \ldots a_n \in \Sigma^n$. If there exists an i such $a_i = v_i$ then the simulator terminates the experiment and outputs abort. Otherwise, the simulator responds with the challenge ciphertext

$$C = (M_b \cdot T,\ \ g_3,\ \ g_3^{\alpha_{1,a_1}},\ \ \ldots,\ \ g_3^{\alpha_{n,a_n}})$$

Suppose $g_3 = g^c$. Then, we note that since $u_{i,a_i} = g^{\alpha_{i,a_i}}$ for all i, we have that:

$$C = \left(M_b \cdot T,\ \ g^c,\ \ u_{1,a_1}^c,\ \ \ldots,\ \ u_{n,a_n}^c\right)$$

Hence, if $T = e(g,g)^{abc} = e(g_1, g_2)^c$ then the challenge C is a valid encryption of M_b under ID^*.

Phase 2. $\mathcal{A}$ issues more private key queries for identities $\mathsf{ID} \neq \mathsf{ID}^*$. The simulator responds as before.

Guess. Finally, $\mathcal{A}$ outputs a guess $b' \in \{0,1\}$. The simulator outputs b' as the result of the experiment.

We define $\mathsf{BDH\text{-}Exp}_{\mathcal{A}}(b, (g, g_1, g_2, g_3, T))$ to be the random variable denoting the simulator's output in the above experiment. It takes one of three values: 0, 1, or abort.

Experiment 2: $\mathsf{PRF\text{-}Exp}_{\mathcal{A}}(b, F, k)$. Let $\mathcal{A}$ be an algorithm, b be a bit in $\{0,1\}$, F be a function $F : \{0,1\}^w \to \{0,1\}$, and $k \in \mathcal{K}$. Define the following game between a simulator and $\mathcal{A}$:

Setup: To generate system parameters the simulator picks a random $\alpha \in \mathbb{Z}_p$ and sets $g_1 = g^\alpha$. Next, it picks random $g_2 \in \mathbb{G}$ and a random $n \times s$ matrix $U = (u_{i,j})$ where each $u_{i,j} \in \mathbb{G}$. It gives $\mathcal{A}$ the system parameters $\mathit{params} = (g, g_1, g_2, U, k)$ and keeps to itself the master key $\mathit{master\text{-}key} = g_2^\alpha$.

Phase 1: $\mathcal{A}$ issues up to q adaptive private key queries. Consider a query for the private key $\mathsf{ID} \in \{0,1\}^w$. If $F(\mathsf{ID}) = 1$ the simulator terminates the experiment and outputs abort. Otherwise, the simulator uses *master-key* to generate the private key for ID and gives the result to $\mathcal{A}$.

Challenge. $\mathcal{A}$ outputs an identity ID^* and two messages $M_0, M_1 \in \mathbb{G}_1$. If $F(\mathsf{ID}) = 0$ the simulator terminates the experiment and outputs abort. Otherwise, the simulator creates the encryption of M_b and gives the resulting challenge ciphertext to $\mathcal{A}$.

Phase 2. $\mathcal{A}$ issues more private key queries for identities $\mathsf{ID} \neq \mathsf{ID}^*$. The simulator responds as before (aborting as necessary).

Guess. Finally, $\mathcal{A}$ outputs a guess $b' \in \{0,1\}$. The simulator outputs b' as the result of the experiment.

We define $\mathsf{PRF\text{-}Exp}_{\mathcal{A}}(b, F)$ to be the random variable denoting the simulator's output in the above experiment. It takes one of three values: 0, 1, or abort.

Next, we state four facts about these experiments, which we prove in the full version of the paper. The proof of Theorem 1 will follow immediately from these facts. We define the following notation:

1. Define the random variable $Z = (g, g_1, g_2, g_3, T) \xleftarrow{R} \mathcal{P}_{\text{BDH}}$.
2. For $b = 0, 1$ define the random variable $T_b = \mathsf{BDH\text{-}Exp}_{\mathcal{A}}(b, Z)$.
3. For $b = 0, 1$ define the value $t_b = \Pr[T_b = 1 \mid T_b \neq \mathsf{abort}]$.
4. We let $\{F_{K,H_k}\}$ denote the distribution sampled by the following algorithm: pick a random $k \in \mathcal{K}$ and a random $K \in \Sigma^{(n,m)}$, and output the (function, key) pair (F_{K,H_k}, k).
5. We set $\delta = (1 - 1/s)^m$ and $\Delta = \delta(1-\delta)^q$.

Claim 1. Consider $(F, k) \xleftarrow{R} \{F_{K,H_k}\}$. Then for $b = 0, 1$ the random variable $T_b = \mathsf{BDH\text{-}Exp}_{\mathcal{A}}(b, Z)$ is identical to the random variable $\mathsf{PRF\text{-}Exp}_{\mathcal{A}}(b, F, k)$.

Claim 2. For $b = 0, 1$ we have that t_b is equal to $\Pr[\mathsf{CPA\text{-}Exp}_{\mathcal{A}}(b) = 1]$.

Claim 3. Let $(F, k) \xleftarrow{R} \{F_{K,H_k}\}$. Then for $b = 0, 1$:

$$\Pr[\mathsf{PRF\text{-}Exp}_{\mathcal{A}}(b, F, k) = \mathsf{abort}] < 1 - \Delta + \epsilon_{\text{PRF}}$$

Claim 4. We have that $|t_0 - t_1| < 2\epsilon_{\text{BDH}}/(\Delta - \epsilon_{\text{PRF}})$.

The proofs of these claims are given in the full version of the paper. The main theorem follows easily.

Proof (Proof of Theorem 1). The theorem follows directly from Claims 2 and 4. The two claims together show that for any t-time algorithm $\mathcal{A}$ that makes at most q private key queries, we have

$$\left| \Pr[\mathsf{CPA\text{-}Exp}_{\mathcal{A}}(0) = 1] - \Pr[\mathsf{CPA\text{-}Exp}_{\mathcal{A}}(1) = 1] \right| = |t_0 - t_1| < 2\epsilon_{\text{BDH}}/(\Delta - \epsilon_{\text{PRF}})$$

as required. □

5 Constructing Admissible Hash Functions

It remains to show how an admissible hash function family can be constructed given a collision resistant hash function family. We do this in two steps: we first present some idealized sufficient conditions for a hash function family to be admissible, then show how these conditions can be achieved in the case of a binary alphabet given a family of collision resistant hash functions. As previously mentioned, the Decision BDH assumption can be used to realize collision resistance, although we are free to use more practical hash functions.

For simplicity, we define the following shorthand notation. We let $\Sigma^{(n,m)}$ be the universe of the possible values of the secret index K. For a hash function H, we respectively define the *H-null-set* and the *H-kernel* of any $x \in \{0,1\}^w$ as:

$$Z_H(x) = \{K \in \Sigma^{(n,m)} : F_{K,H}(x) = 0\}, \quad Y_H(x) = \{K \in \Sigma^{(n,m)} : F_{K,H}(x) = 1\}$$

Clearly, for any x the sets $Z_H(x)$ and $Y_H(x)$ form a partition of $\Sigma^{(n,m)}$ such that $|Z_H(x)| = \binom{n}{m}(s^m - (s-1)^m)$ and $|Y_H(x)| = \binom{n}{m}(s-1)^m$. For binary alphabets, we have

$$(s = 2) \;\Rightarrow\; |\Sigma^{(n,m)}| = \binom{n}{m} 2^m, \quad |Z_H(x)| = \binom{n}{m}(2^m - 1), \quad |Y_H(x)| = \binom{n}{m}$$

Before delving into the construction, we need to precise the following notions.

Adversarial Uncertainty. We formalize the information made available to the adversary using the notion of knowledge state. At any time during the interaction of an algorithm $\mathcal{A}^F$ with a bias map oracle $F_{K,H}$ where H is public and K is secret, the algorithm's available knowledge about the oracle is captured by a distribution of the secret K. Initially the distribution is uniform over $\Sigma^{(n,m)}$ since K is chosen uniformly in this set. Now, suppose that prior to the next interaction with the oracle the distribution is uniform over some set S, then the distribution after the next oracle query $F_{K,H}(x_i)$ is uniform over a subset $S' \subseteq S$ such that

$$S' = \begin{cases} S \cap Z_H(x) & \text{if } F_{K,H}(x_i) = 0 \\ S \cap Y_H(x) & \text{if } F_{K,H}(x_i) = 1 \end{cases}$$

It follows that after learning the responses $\{F_{K,H}(x_i) : i = 1, \ldots, j\}$ to any set of queries $\{x_i : i = 1, \ldots, j\}$, the algorithm's knowledge state regarding K is completely captured by the uniform distribution over the set S_j given by

$$S_j \;=\; \left(\Sigma^{(n,m)}\right) \;\cap\; \underbrace{\bigcap_{\substack{i \in \{1,\ldots,j\} \\ F_{K,H}(x_i)=0}} Z_H(x_i)}_{S_j^{(0)}} \;\cap\; \underbrace{\bigcap_{\substack{i \in \{1,\ldots,j\} \\ F_{K,H}(x_i)=1}} Y_H(x_i)}_{S_j^{(1)}}$$

Here, $S_j^{(0)}$ and $S_j^{(1)}$ are respectively defined as the sets of values of $K \in \Sigma^{(n,m)}$ that are compatible with the "negative" and the "positive" responses from the set of oracle responses $\{F_{K,H}(x_i) : i = 1, \ldots, j\}$. Notice that reordering the queries has no effect on the knowledge state.

Hamming Separation Property. For two vectors $x, y \in \Sigma^n$, we write $d(x, y)$ for the Hamming distance between x and y. We say that a hash function family $\{H_k : \{0,1\}^w \to \Sigma^n\}_{k \in \mathcal{K}}$ satisfies the *v-Hamming separation* property if $\forall k \in \mathcal{K}$ and $\forall x, y \in \{0,1\}^w$ such that $H_k(x) \neq H_k(y)$, it also holds that $d(H_k(x), H_k(y)) \geq v$. In other words, any distinct $H_k(x)$ and $H_k(y)$ must take differing values in at least v coordinates (and thus have at most $n - v$ coordinates in common).

In Section 5.2, we show how to achieve the Hamming separation property from collision resistance using coding theory.

5.1 Sufficient Conditions For Admissibility

The following theorem gives a set of sufficient conditions for a hash family to be admissible as defined in Definition 6. We focus on binary alphabets ($s = 2$).

Theorem 2. *Let n, m, v, w be positive integers such that $m \leq n$ and $v \leq n$. Let Σ be an alphabet of size $s = 2$, and let $\delta = (1 - 1/s)^m = 2^{-m}$. Assume that $\mathcal{H} = \{H_k : \{0,1\}^w \to \Sigma^n\}_{k \in \mathcal{K}}$ is some (t, ϵ_H)-collision resistant hash function family that satisfies the v-Hamming separation property. Pose $\theta = (1 - v/n)^m$. If $\theta \leq \kappa\delta$ for some arbitrary $\kappa \in (1, \infty)$ then the family $\mathcal{H}$ is $(t, \epsilon_{PRF}, q, m)$-admissible provided that $\epsilon_{PRF} \geq \epsilon_H + \frac{13}{2}\gamma^2/\kappa$ and $q \leq \gamma/\kappa\delta$ for some arbitrary $\gamma \in (0, \frac{1}{2})$.*

Proof. It suffices to show that, in the view of any algorithm $\mathcal{A}$ interacting with a bias map oracle F_{K,H_k} for random $k \in \mathcal{K}$ and $K \in \Sigma^{(n,m)}$ where K is secret, the first q outputs of the oracle are distributed identically to the first q outcomes of a binomial random process of expectation δ, with probability at least $1 - \epsilon_{\mathrm{PRF}}$.

We henceforth omit the subscripts K and H_k since there is no ambiguity, and write $F(x)$ for $F_{K,H_k}(x)$. We use the abbreviations $Y_i = Y_{H_k}(x_i)$, $Z_i = Z_{H_k}(x_i)$, $h_i = H_k(x_i)$, and $F_i = F(x_i)$.

We compute the distribution of the first q oracle answers under the stated assumptions, treating the algorithm $\mathcal{A}$ as an adversary that adaptively selects the q points $x_1, \ldots, x_q$ at which F is queried. For now, we assume that $\forall i \neq j : x_i \neq x_j \Rightarrow h_i \neq h_j$ (and by the v-Hamming separation property, $d(h_i, h_j) \geq v$). By the $(t, \epsilon_{\mathrm{H}})$-collision resistance assumption on $\mathcal{H}$, this is true with probability at least $1 - \epsilon_{\mathrm{H}}$. We correct for this assumption at the end.

Suppose that before step $j \in \{1, \ldots, q\}$ the adversary has learned the $j-1$ values respectively taken by $F(x)$ at arbitrary query points $x = x_1, \ldots, x_{j-1}$. Our goal is to find lower and upper bounds on the conditional probability that $F(x_j) = 1$ given the history of past queries and answers, in the adversary's view, uniformly for all choices of the next query point $x_j \notin \{x_1, \ldots, x_{j-1}\}$.

Let $X_i = \{x_1, \ldots, x_i\} = X_i^{(0)} \cup X_i^{(1)}$ where $X_i^{(0)} = \{x \in X_i : F(x) = 0\}$ and $X_i^{(1)} = \{x \in X_i : F(x) = 1\}$, and write $P_j = \Pr[F(x_j) = 1 \mid X_{j-1}^{(0)}, X_{j-1}^{(1)}]$ for the probability we seek to bound. Observe that the two sets $X_{j-1}^{(0)}$ and $X_{j-1}^{(1)}$ together capture all relevant information about the query history just before the j-th query, since the order of the queries is irrelevant. We have

$$P_j = \Pr[F(x_j) = 1 \mid X_{j-1}^{(0)}, X_{j-1}^{(1)}] = \frac{|Y_j \cap S_{j-1}|}{|S_{j-1}|} = \frac{|Y_j \cap S_{j-1}^{(1)} \cap S_{j-1}^{(0)}|}{|S_{j-1}^{(1)} \cap S_{j-1}^{(0)}|}$$

$$= \frac{\left|Y_j \cap \left(\bigcap_{x \in X_{j-1}^{(1)}} Y(x)\right) \cap \left(\bigcap_{x \in X_{j-1}^{(0)}} Z(x)\right)\right|}{\left|\left(\bigcap_{x \in X_{j-1}^{(1)}} Y(x)\right) \cap \left(\bigcap_{x \in X_{j-1}^{(0)}} Z(x)\right)\right|} = \frac{\left|Y_j \cap Y_{\cap,j-1}^{(1)} \setminus Y_{\cup,j-1}^{(0)}\right|}{\left|Y_{\cap,j-1}^{(1)} \setminus Y_{\cup,j-1}^{(0)}\right|}$$

where we have posed $Y_{\cap,j-1}^{(1)} = \left(\cap_{x \in X_{j-1}^{(1)}} Y(x)\right)$ and $Y_{\cup,j-1}^{(0)} = \left(\cup_{x \in X_{j-1}^{(0)}} Y(x)\right)$.

We can use this general expression and the v-Hamming separation property to bound P_j for query histories that contain either zero or one positive answer. We later show that the other cases are together very unlikely. Namely, we seek:

1. a uniform bounding interval on P_j for all query histories with $|X_{j-1}^{(1)}| = 0$ (i.e., containing only negative answers);
2. a uniform upper bound on P_j for all query histories such that $|X_{j-1}^{(1)}| = 1$ (i.e., containing one positive answer).

We obtain non-trivial uniform bounds of three different kinds, given by

$$\begin{array}{lll} \forall X_{j-1}^{(0)}, X_{j-1}^{(1)} \text{ s.t. } |X_{j-1}^{(1)}| = 0 : & (1-\gamma)\delta \leq P_j \leq (1+2\gamma)\delta \\ \forall X_{j-1}^{(0)}, X_{j-1}^{(1)} \text{ s.t. } |X_{j-1}^{(1)}| = 1 : & P_j \leq 2\kappa\,\delta \end{array}$$

Detailed calculations for these bounds are given in the full version of the paper.

Subject to the above inequalities, we set out to bound the probability that the biased PRF oracle F deviates from a sequence of q outcomes from a genuine memoryless binomial process of expectation δ over a sequence of length q.

Consider R, a binomial process of expectation δ. We construct a modified process R' whose i-th outcome is defined as $R'_i = R_i \oplus M_i$. Here, M is a control process whose purpose is to randomly decide whether R'_i should assume the value of R_i or its opposite, with a probability that depends on the previous outcomes $R'_1, \ldots, R'_{i-1}$ and the current drawing R_i. By properly choosing M, we can make R' behave exactly as F, i.e., have the q-prefixes of R' achieve the same joint distribution as the q-prefix of F. In particular, this means that the event that the processes R and F behave similarly over a sequence of length q is at least as likely as the event that $M_i = 0$ for all $i = 1, \ldots, q$, since in this case R and R' have the same first q outcomes. It remains to bound such probability. Here is the gist of the argument.

The goal is to devise an R' that perfectly simulates any q-prefix of $F = F_{K,H}$ for (unknown) random K, and bound the influence of M needed to do so. Suppose that for some query history $X^{(0)}_{j-1}, X^{(1)}_{j-1}$, the conditional expectation $P_j = \Pr[F_j = 1 \mid X^{(0)}_{j-1}, X^{(1)}_{j-1}]$ of F_j as viewed by the adversary exceeds the expectation $\Pr[R_j = 1] = \delta$ of the binomial process R_j. One can make the simulated process R'_j assume the expected law of F_j conditionally on this specific history by letting the control process take $M_j \leftarrow 1$ with conditional probability $(P_j - \delta)/(1 - \delta)$ when $R_i = 0$, and with probability 0 when $R_i = 1$. More generally, we find that for the process R' to perfectly simulate F, it suffices that for $j = 1, \ldots, q$, the conditional law of M_j given $R'_1, \ldots, R'_{j-1}, R_j$ satisfies

$$\Pr[M_j = 1, R_j = 0 \mid X^{(0)}_{j-1}, X^{(1)}_{j-1}] = \max\{\, 0,\ \ P_j - \delta \,\}$$
$$\Pr[M_j = 1, R_j = 1 \mid X^{(0)}_{j-1}, X^{(1)}_{j-1}] = \max\{\, 0,\ \ \delta - P_j \,\}$$

Let us write E_j for the event $[\exists i \leq j, M_i \neq 0]$. We outline how to use the above results to upper bound the unconditional probability $\Pr[E_j]$ for $j \leq q$. First, from the law of M we get $\Pr[M_j = 1 \mid X^{(0)}_{j-1}, X^{(1)}_{j-1}] \leq |P_j - \delta| \leq 1$, which we can bound further using our previous bounds on P_j in the cases where $|X^{(1)}_{j-1}| = 0, 1$. Next, we need to bound the probabilities $\Pr[X^{(0)}_{j-1}, X^{(1)}_{j-1}]$ of the conditioning events. The difficultly here is that the random variables $X^{(0)}_{j-1}, X^{(1)}_{j-1}$ derive from the complicated process R'. Fortunately, conditionally on the event $\neg E_{j-1}$, the process R' identifies with the binomial process R so that these probabilities have nice expressions in function of j and $|X^{(1)}_{j-1}|$. Note that these probabilities vanish quickly as $|X^{(1)}_{j-1}|$ increases, which is why we bounded P_j for $|X^{(1)}_{j-1}| = 0, 1$ only.

Thus, we have just reduced the upper bound computation of $\Pr[E_j]$ to that of $\Pr[E_{j-1}]$. Carrying this idea through, after some calculations we obtain

$$\Pr[E_q] \;=\; \Pr[\exists i \leq j, M_i \neq 0] \;=\; \sum_{j=1}^{q} \Pr[M_j = 1, \neg E_{j-1}] \;\leq\; \frac{13}{2}\gamma^2/\kappa$$

The formal derivation of this result may be found in the full version of the paper.

To conclude, we correct for the probability ϵ_{H} of finding a hash collision in the allotted time t, which in the worst scenario could yield an infallible discriminator between F and R. It follows that the probability that the F and R oracles can be distinguished admits the upper bound $\epsilon_{\mathrm{H}} + \frac{13}{2}\gamma^2/\kappa \leq \epsilon_{\mathrm{PRF}}$, as required. $\square$

5.2 Admissibility From Collision Resistance

We now show how to construct an admissible hash function family $\mathcal{H} = \{H_k : \{0,1\}^w \to \Sigma^n\}_{k\in\mathcal{K}}$ in the sense of Theorem 2, given an "ordinary" family of $(t, \epsilon_{\mathrm{H}})$-collision resistant hash functions $\bar{\mathcal{H}} = \{\bar{H}_k : \{0,1\}^w \to \{0,1\}^\beta\}_{k\in\mathcal{K}}$. We give an explicit construction for the specific case of a binary alphabet ($s = 2$).

Theorem 3. *Let $\bar{\mathcal{H}} = \{\bar{H}_k : \{0,1\}^w \to \{0,1\}^\beta\}_{k\in\mathcal{K}}$ be an efficiently computable (t, ϵ_H)-collision resistant hash function family. Then for any $r \in (0, \frac{1}{2})$ there exists an efficiently computable function family $\mathcal{H} = \{H_k : \{0,1\}^w \to \{0,1\}^n\}_{k\in\mathcal{K}}$ that satisfies both the (t, ϵ_H)-collision resistance property and the bitwise v-Hamming separation property, where $\beta \leq n \leq 2\beta^2/(1-2r)^2$ and $v/n > r$.*

Proof. Let t be the smallest positive integer such that $2^t \geq \lceil \beta/t \rceil/(1-2r) + 1$, and define $\ell = \lceil \beta/t \rceil$.

Let $\mu' : \{0,1\}^t \to \mathbb{F}_{2^t}$ be any bijection. Define the injection $\mu : \{0,1\}^\beta \to \mathbb{F}_{2^t}^\ell$ that, on input $z \in \{0,1\}^\beta$, partitions z in ℓ fragments of t bits each (padding the last fragment as necessary), applies the map μ' to each fragment, and concatenates all the outputs.

Let $\rho : \mathbb{F}_{2^t}^\ell \to \mathbb{F}_{2^t}^{2^t-1}$ be a Reed-Solomon error correcting code with parameters $[2^t-1, \ell, 2^t-\ell]$, i.e., a linear code that takes input words of size ℓ over the alphabet $\mathbb{F}_{2^t}$ and produces codewords of length $2^t - 1$ with minimum pairwise Hamming distance $2^t - \ell$.

Let $\eta' : \mathbb{F}_{2^t} \to \{0,1\}^{2^t}$ be the injection that maps any field element $i \in \{0, \ldots, 2^t-1\}$ to the 2^t-bit vector given by the i-th row of a $2^t \times 2^t$ Hadamard matrix. Recall that a binary $m \times m$ Hadamard matrix is such that any two distinct rows or columns agree on exactly $m/2$ coordinates; it is well known that a $2^t \times 2^t$ Hadamard matrix exists and is easy to construct for all $t \geq 1$. Define the function $\eta : \mathbb{F}_{2^t}^{2^t-1} \to \{0,1\}^{2^t(2^t-1)}$ that applies η' individually to each coordinate of its input word and concatenates the resulting Hadamard vectors.

The desired hash family is then given by $\mathcal{H} = \{H_k : \{0,1\}^w \to \Sigma^n\}_{k\in\mathcal{K}}$ where $H_k = \eta \circ \rho \circ \mu \circ \bar{H}_k$.

It remains to show that $\mathcal{H}$ has the desired properties.

First, since $\eta \circ \rho \circ \mu$ is an injection, the $(t, \epsilon_{\mathrm{H}})$-collision resistance of $\bar{H}_k$ entails the same for H_k.

Next, by the stated properties of the Reed-Solomon code, ρ produces codewords of size $2^t - 1$ with minimum pairwise Hamming distance $2^t - \ell$ in $\mathbb{F}_{2^t}$. Since η turns any two distinct elements of $\mathbb{F}_{2^t}$ into 2^t-bit vectors that differ in 2^{t-1}

positions, it follows that $\eta \circ \rho$ produces binary vectors of size $n = 2^t(2^t - 1)$ with minimum pairwise Hamming distance $v = 2^{t-1}(2^t - \ell)$ in $\mathbb{F}_2$. The corresponding ratio v/n is bounded as follows. Since t is chosen such that $(2^t - 1)(1 - 2r) \geq \ell$, we have $2^t - \ell \geq 2r(2^t - 1) + 1$, hence $(2^t - \ell)/(2^t - 1) > 2r$. It follows that $v/n > r$, as claimed.

Last, we have that $\beta \leq n = 2^t(2^t - 1) \leq 2\lceil \beta/t \rceil^2/(1-2r)^2 \leq 2\,\beta^2/(1-2r)^2$, as required. □

5.3 Putting It All Together—Concrete Bounds

It is useful to assign a more concrete meaning to the values taken by the parameters intervening in Theorems 2 and 3. We assume to be given ϵ_{H} (the adversarial advantage against the collision resistant hash functions), β (the collision resistant hash output length in bits), and q (the allowed number of PRF queries), under the "birthday paradox" constraint that $q \ll \sqrt{2^\beta}$. Our task is to find a suitable set of parameters so that (1) the security ϵ_{IBE} of the IBE system of Section 4.2 is within a polynomial factor of ϵ_{BDH}, and (2) the complexity of the four IBE operations takes polynomial time in the security parameters. For $s = 2$, we require that $\epsilon_{\mathrm{IBE}}/\epsilon_{\mathrm{BDH}} \leq O(\mathrm{poly}(q))$ and $n \leq O(\mathrm{poly}(\beta, \log(q), \log(1/\epsilon_{\mathrm{IBE}}))$.

We describe two settings of the parameters; one favoring security, the other favoring performance.

Favoring security. We first show how to satisfy the requirements for the PRF construction with a binary alphabet ($s = 2$) when the intrinsic PRF error probability (defined as $\epsilon'_{\mathrm{PRF}} = \epsilon_{\mathrm{PRF}} - \epsilon_{\mathrm{H}}$ in the notation of Theorem 2) is pegged to $\epsilon'_{\mathrm{PRF}} = \epsilon_{\mathrm{H}}$. We arbitrarily choose $\kappa \leftarrow 2$ and successively derive: $\gamma \leftarrow \sqrt{\epsilon'_{\mathrm{PRF}}}/2$, $m \leftarrow \lfloor \log_2(2q/\gamma) \rfloor$, $\delta \leftarrow 2^{-m} \approx \gamma/2q$, $r \leftarrow 1 - \sqrt[m]{2}/2 \approx \frac{1}{2} - \frac{1}{3m}$, $t \leftarrow$ least s.t. $2^t \geq \lceil \beta/t \rceil/(1-2r) + 1$, $\ell \leftarrow \lceil \beta/t \rceil$, $n \leftarrow 2^t(2^t - 1)$, $v \leftarrow 2^{t-1}(2^l - \ell)$, and $\theta \leftarrow (1 - \frac{v}{n})^m < \kappa\,\delta$. Evidently, the total PRF loss $\epsilon_{\mathrm{PRF}} = \epsilon_{\mathrm{H}} + \epsilon'_{\mathrm{PRF}} = 2\epsilon_{\mathrm{H}}$ is negligible and the bandwidth coefficient $n = O(\log_2^2(q/\sqrt{\epsilon_{\mathrm{PRF}}})\beta^2)$ is polynomial in $\log q$ and β. The price to pay for such a low value of ϵ_{PRF} is a fairly large n.

Favoring performance. We can attain better bounds by adjusting the PRF loss to best match the intrinsic loss incurred by the IBE construction itself, in function of q, as follows. Assuming that the loss ϵ_{H} due to hash collisions is negligible, under the $(t, \epsilon_{\mathrm{BDH}})$-Decision BDH assumption Theorem 1 gives a $(t, q, \epsilon_{\mathrm{IBE}})$-secure IBE such that $\epsilon_{\mathrm{IBE}} = 2\,\epsilon_{\mathrm{BDH}}/(\delta(1-\delta)^q - \epsilon_{\mathrm{PRF}}) \approx 2\,\epsilon_{\mathrm{BDH}}/(\sqrt{\epsilon_{\mathrm{PRF}}}/4q - \epsilon_{\mathrm{PRF}})$. We can minimize ϵ_{IBE} for a prescribed value of q by seeking $\epsilon_{\mathrm{PRF}} \leftarrow (1/8q)^2$. For $\kappa \leftarrow 2$ this gives us a total IBE security loss $\epsilon_{\mathrm{IBE}} \approx 64\,q^2\,\epsilon_{\mathrm{BDH}} = \Theta(q^2 \epsilon_{\mathrm{BDH}})$ under the improved bandwidth requirement $n \leq (9 + 4\log_2 q)^2 \beta^2 = \Theta((\log_2 q)\beta^2)$.

We note that the optimal value of κ varies and is tied to the coding construction. We defer to the full paper the question of optimizing for all parameters.

6 Extensions

We very briefly outline a few simple extensions of the IBE system of Section 4.2.

Hierarchical IBE. Introduced by Horowitz and Lynn [HL02], HIBE was first constructed by Gentry and Silverberg [GS02] in the random oracle model. The IBE system of Section 4.2 generalizes naturally to give a semantically secure HIBE under an adaptive chosen identity attack ($\mathsf{IND\text{-}ID\text{-}CPA}$) without random oracles. For a hierarchy of depth ℓ, both the ciphertext and private key contain ℓ blocks where each block contains n components. Thus, a private key at depth ℓ is an element of $\mathbb{G}^{\ell n+1}$. As our IBE, the HIBE uses collision resistant hash functions and is provably secure without random oracles whenever the Decision BDH assumption holds. The construction is similar to the construction of a (selective identity secure) HIBE without random oracles based on Decision BDH recently proposed by Boneh and Boyen [BB04]. The details are deferred to the full version of the paper.

Chosen Ciphertext Security. A recent result of Canetti et al. [CHK04] gives an efficient way to build a chosen ciphertext IBE ($\mathsf{IND\text{-}ID\text{-}CCA}$) from a chosen plaintext 2-HIBE ($\mathsf{IND\text{-}ID\text{-}CPA}$). Thus, by the previous paragraph, we obtain a full chosen identity, chosen ciphertext IBE ($\mathsf{IND\text{-}ID\text{-}CCA}$) that is provably secure without random oracles. More generally, by starting from an $(\ell+1)$-HIBE, a fully secure ℓ-HIBE can be similarly constructed without random oracles.

Arbitrary Identities. We can extend our IBE system to handle identities $\mathsf{ID} \in \{0,1\}^*$ (as opposed to $\mathsf{ID} \in \{0,1\}^w$) by first hashing ID using a collision resistant hash function $\tilde{H} : \{0,1\}^* \to \{0,1\}^w$ prior to key generation and encryption. A standard argument shows that if the scheme of Section 4.2 is $\mathsf{IND\text{-}ID\text{-}CPA}$ secure then so is the scheme with the additional hash. This holds for the HIBE and the chosen ciphertext secure system and as well.

7 Conclusions

We presented an Identity Based cryptosystem and proved its security without using the random oracle heuristic under the decisional Bilinear Diffie-Hellman assumption. Our results prove that secure IBE systems exist in the standard model. This resolves an open problem posed by Boneh and Franklin in 2001. However, the present system is not very practical and mostly serves as an existence proof. It is still a wonderful problem to find a practical IBE system secure without random oracles based on Decision BDH or a comparable assumption.

References

[BB04] Dan Boneh and Xavier Boyen. Efficient selective-ID identity based encryption without random oracles. In *Advances in Cryptology—EUROCRYPT '04*, volume 3027 of *LNCS*, pages 223–238. Springer-Verlag, 2004.

[BBP04] Mihir Bellare, Alexandra Boldyreva, and Adriana Palacio. An uninstantiable random-oracle-model scheme for a hybrid-encryption problem. In *Advances in Cryptology—EUROCRYPT '04*, volume 3027 of *LNCS*, pages 171–188. Springer-Verlag, 2004.

[BF01] Dan Boneh and Matt Franklin. Identity-based encryption from the Weil pairing. In *Advances in Cryptology—CRYPTO '01*, volume 2139 of *LNCS*, pages 213–229. Springer-Verlag, 2001.

[BF03] Dan Boneh and Matt Franklin. Identity-based encryption from the Weil pairing. *SIAM J. of Computing*, 32(3):586–615, 2003.

[Boy03] Xavier Boyen. Multipurpose identity-based signcryption: A Swiss Army knife for identity-based cryptography. In *Advances in Cryptology—CRYPTO '03*, volume 2729 of *LNCS*, pages 383–99. Springer-Verlag, 2003.

[BR93] Mihir Bellare and Phil Rogaway. Random oracle are practical: A paradigm for designing efficient protocols. In *Proceedings of ACM Conference on Computer and Communications Security—CCS '93*, pages 62–73, 1993.

[CC03] Jae Choon Cha and Jung Hee Cheon. An identity-based signature from gap Diffie-Hellman groups. In *Practice and Theory in Public Key Cryptography—PKC '03*, volume 2567 of *LNCS*. Springer-Verlag, 2003.

[CGH98] Ran Canetti, Oded Goldreich, and Shai Halevi. The random oracle model revisited. In *Proceedings of ACM Symposium on Theory of Computing—STOC '98*. ACM, 1998.

[CHK03] Ran Canetti, Shai Halevi, and Jonathan Katz. A forward-secure public-key encryption scheme. In *Advances in Cryptology—EUROCRYPT '03*, volume 2656 of *LNCS*, pages 255–271. Springer-Verlag, 2003.

[CHK04] Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. In *Advances in Cryptology—EUROCRYPT '04*, volume 3027 of *LNCS*, pages 207–222. Springer-Verlag, 2004.

[Coc01] Clifford Cocks. An identity based encryption scheme based on quadratic residues. In *Proceedings of the 8th IMA International Conference on Cryptography and Coding*, pages 26–28, 2001.

[GS02] Craig Gentry and Alice Silverberg. Hierarchical ID-based cryptography. In *Advances in Cryptology—ASIACRYPT '02*, volume 2501 of *LNCS*, pages 548–566. Springer-Verlag, 2002.

[HK04] Swee-Huay Heng and Kaoru Kurosawa. k-resilient identity-based encryption in the standard model. In *Topic in Cryptology—CT-RSA '04*, volume 2964 of *LNCS*, pages 67–80, 2004.

[HL02] Jeremy Horwitz and Benjamin Lynn. Towards hierarchical identity-based encryption. In *Advances in Cryptology—EUROCRYPT '02*, pages 466–481, 2002.

[Jou00] Antoine Joux. A one round protocol for tripartite Diffie-Hellman. In *Proceedings of ANTS IV*, volume 1838 of *LNCS*, pages 385–394. Springer-Verlag, 2000.

[MY96] Ueli M. Maurer and Yacov Yacobi. A non-interactive public-key distribution system. *Designs, Codes and Cryptography*, 9(3):305–316, November 1996.

[Sha84] Adi Shamir. Identity-based cryptosystems and signature schemes. In *Advances in Cryptology—CRYPTO '84*, volume 196 of *LNCS*, pages 47–53. Springer-Verlag, 1984.

[SOK00] Ryuichi Sakai, Kiyoshi Ohgishi, and Masao Kasahara. Cryptosystems based on pairings. In *Proceedings of Symposium on Cryptography and Information Security—SCIS '00*, Japan, 2000.

[Tan87] Hatsukazu Tanaka. A realization scheme for the identity-based cryptosystem. In *Advances in Cryptology—CRYPTO '87*, volume 293 of *LNCS*, pages 341–349. Springer-Verlag, 1987.