An extended abstract of this paper is to appear in Advances in Cryptology—CRYPTO 2004, Springer-Verlag. This version is available from the cryptology eprint archive at `http://eprint.iacr.org/2004/173/`.

Secure Identity Based Encryption Without Random Oracles

Dan Boneh*
dabo@cs.stanford.edu

Xavier Boyen†
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 decisional bilinear Diffie-Hellman assumption. Previous constructions of this type incurred a large penalty factor in the security reduction from the underlying complexity assumption. The security reduction of the present system is polynomial in all the parameters.

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 the identity based encryption model was proposed two decades ago [Sha84], and a few early precursors suggested over the years [Tan87, TI89, 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 (BDH) 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 only 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 recent uninstantiable random oracle cryptosystems [CGH98, BBP04], which are secure in the random oracle model, but are provably insecure under any actual instantiation of the oracle. Towards this goal, several recent results [CHK03, BB04a, HK04] construct IBE systems secure without random oracles in weaker versions of the Boneh-Franklin model. In one such model, called "selective-ID" IBE [CHK03], the adversary must commit ahead of time to the identity it wishes to attack.

It is easy to show that any selective-ID secure IBE is readily converted into a fully secure IBE by artificially restricting the space of identities, but the proof uses an inefficient security reduction [BB04b, §7]. For example, if all identities in the restricted scheme can be represented as n-bit binary strings, then the reduction degrades security by a factor of 2^n. Concretely, suppose that identities in the system are encoded in 160 bits, the length of SHA-1 digests. Further, suppose that one of the selective-ID IBE systems of [CHK03, BB04a] is instantiated with a sufficiently large

*Computer Science Dept., Stanford University. Supported by NSF and the Packard Foundation.
†Voltage Security Inc., Palo Alto, California.

security parameter that no t-time adversary has advantage 2^{-240} in a selective-ID attack. Then, according to [BB04b, §7], no t-time adversary can have advantage 2^{-80} in a full adaptive identity attack against the same system with the 160-bit restriction on identities. This restriction is not limiting in any way since one can always hash long indentites with SHA-1. In other words, these IBE systems are fully secure in the sense of Boneh-Franklin in the standard model, provided that a sufficiently large bilinear group is used. Unfortunately, as mentioned above, the generic reduction from selective-ID to full adaptive-ID security is not polynomial time.

In view of this, a natural question is whether a fully secure IBE can be built with a polynomially bounded reduction from the underlying complexity assumption. In this paper we construct such a cryptosystem. Security is based on the decisional version of the bilinear Diffie-Hellman assumption. Our construction demonstrates that fully secure IBE systems with a polynomial time reduction can exist in the absence of random oracles. The main shortcoming of the proposed system is that it is impractical; consequently, we mostly view our construction as an existence proof. This contrasts with the two selective identity constructions from [BB04a], which are very simple and practical even when scaled for full IBE security.

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 $x \xleftarrow{\text{R}} S$ to define a random variable x that picks an element of S uniformly at random. For a randomized algorithm $\mathcal{A}$ we use $x \xleftarrow{\text{R}} \mathcal{A}(y)$ 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 secret *master-key*. The *KeyGen* algorithm uses the master secret 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 either a private key or a decryption query. These queries may be asked adaptively, that is, each query q_i may depend on the replies to $q_1, \ldots, q_{i-1}$. The two types of queries are as follows:

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

Decryption query on a ciphertext C_i for an identity ID_i: The challenger responds by executing 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.

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^*. In response, the challenger assembles a ciphertext $C^* = \mathit{Encrypt}(\mathit{params}, \mathsf{ID}^*, M_b)$. It submits the ciphertext C^* as challenge to $\mathcal{A}$.

Phase 2: Algorithm $\mathcal{A}$ issues additional queries $q_{m+1}, \ldots, q_n$, which can be asked adaptively as in Phase 1. Each Phase 2 query q_i is of one of two types:

Private key generation query for any identity ID_i where $\mathsf{ID}_i \neq \mathsf{ID}^*$: The challenger responds as to a Phase 1 query to generate a private key for ID_i.

Decryption query for identity ID^* on a ciphertext C_i with $C_i \neq C^*$: The challenger responds as to a Phase 1 query to decrypt C_i for identity ID^*.

Guess: Eventually, $\mathcal{A}$ concludes the game and 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 2.1. We say that an IBE system $\mathcal{E}$ is $(t, q_{\mathsf{ID}}, q_C, \epsilon_{\mathrm{IBE}})$-adaptive chosen ciphertext secure under a chosen identity attack if for any t-time $\mathsf{IND\text{-}ID\text{-}CCA}$ adversary $\mathcal{A}$ that makes at most q_{ID} chosen private key queries and at most q_C chosen decryption queries we have that $\mathrm{Adv}_{\mathcal{E},\mathcal{A}} < \epsilon_{\mathrm{IBE}}$. As shorthand, we say that $\mathcal{E}$ is $(t, q_{\mathsf{ID}}, q_C, \epsilon_{\mathrm{IBE}})$-$\mathsf{IND\text{-}ID\text{-}CCA}$ 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 2.2. We say that an IBE system $\mathcal{E}$ is $(t, q_{\mathsf{ID}}, \epsilon_{\mathrm{IBE}})$-chosen plaintext secure under a chosen identity attack if $\mathcal{E}$ is $(t, q_{\mathsf{ID}}, 0, \epsilon_{\mathrm{IBE}})$-chosen ciphertext secure under a chosen identity attack. As shorthand, we say that $\mathcal{E}$ is $(t, q_{\mathsf{ID}}, \epsilon_{\mathrm{IBE}})$-$\mathsf{IND\text{-}ID\text{-}CPA}$ 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 (or pairings) and bilinear map groups. Throughout this paper, we let $\mathbb{G}, \mathbb{G}_1, g, e$ be such that:

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

Specifically, for two groups $\mathbb{G}$ and $\mathbb{G}_1$ as above, a bilinear pairing is a map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_1$ with the following properties:

1. bilinearity: $\forall u, v \in \mathbb{G}$, $\forall a, b \in \mathbb{Z}$, $e(u^a, v^b) = e(u, v)^{ab}$;
2. non-degeneracy: $e(g, g) \neq 1$.

Note that $e(\cdot, \cdot)$ is symmetric since $e(g^a, g^b) = e(g, g)^{ab} = e(g^b, g^a)$. Henceforth, for a prime order group $\mathbb{G}$ we denote by $\mathbb{G}^*$ the set $\mathbb{G} \setminus \{1_{\mathbb{G}}\}$ where $1_{\mathbb{G}}$ is the identity element in $\mathbb{G}$; this is the set of generators of $\mathbb{G}$.

We say that $\mathbb{G}$ is a bilinear group if the group operation in $\mathbb{G}$ can be computed efficiently, and there exists a group $\mathbb{G}_1$ and an efficiently computable bilinear pairing $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_1$ as above.

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 the BDH problem 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 space is defined over the random choice of generator $g \in \mathbb{G}^*$, the random choice of exponents $a, b, c \in \mathbb{Z}_p$, and the random bits consumed by $\mathcal{A}$.

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

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

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

- $\mathcal{P}_{\text{BDH}}$ for the distribution over the 5-tuples $\langle g, g^a, g^b, g^c, e(g, g)^{abc}\rangle$ in the left term of (1);
- $\mathcal{R}_{\text{BDH}}$ for the distribution over the 5-tuples $\langle g, g^a, g^b, g^c, T\rangle$ in the right term of (1).

Definition 3.1. We say that the $(t, \epsilon_{\text{BDH}})$-(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:

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

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:

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

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

Definition 3.2. 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 $(\delta, t, \epsilon_{\text{PRF}}, q)$-biased-PRF if for any t-time oracle algorithm $\mathcal{A}$ making at most q queries to its oracle we have

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

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 3.3. 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

$$\Pr\left[H_k(x) = H_k(y) \text{ and } x \neq y \ : \ k \xleftarrow{\text{R}} K;\ (x,y) \xleftarrow{\text{R}} \mathcal{A}(k)\right] < \epsilon_{\text{H}}$$

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 $0 \le m \le n$, 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 $n \ge m > 0$, 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

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

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 4.1. Let n, m, w be positive integers with $m \le n$. 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 $(t, \epsilon_{\text{PRF}}, q, m)$-admissible if the function family $\{F_{K,H_k}\}_{K\in\Sigma^{(n,m)},k\in\mathcal{K}}$ is a $(\delta, t, \epsilon_{\text{PRF}}, q)$-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 inspired from a recent hierarchical IBE construction by Boneh and Boyen [BB04a] with two desirable properties: (i) a tight security reduction without random oracles in the selective-ID attack model; and (ii) a natural indifference to the hierarchical order—which needed to be countered in [BB04a] but that we will now exploit.

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, where the security parameter determines the size 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 $\Sigma = \{1,\ldots,s\}$ 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$(\mathbb{G}, \mathbb{G}_1, e)$: To generate system parameters, the algorithm selects a random generator $g \in \mathbb{G}^*$, picks a random $\alpha \in \mathbb{Z}_p$, and sets $g_1 = g^\alpha$. Next, it picks a random element $g_2 \in \mathbb{G}$ and constructs a random $n \times s$ matrix $U = (u_{i,j}) \in \mathbb{G}^{n\times s}$ where each $u_{i,j}$ is uniform in $\mathbb{G}$. Finally, the algorithm picks a random $k \in \mathcal{K}$ as a hash function key. The system parameters *params* and the master secret *master-key* are given by

$$\textit{params} = \Big(g, g_1, g_2, U, k\Big) \qquad\qquad \textit{master-key} = g_2^\alpha$$

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

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

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

$$C = \left(e(g_1, g_2)^t \cdot M, \quad g^t, \;\; u_{1,a_1}^t, \;\; \ldots, \;\; u_{n,a_n}^t \right) \in \mathbb{G}_1 \times \mathbb{G}^{n+1}$$

Note that $e(g_1, g_2)$ can be precomputed once and for all, or included in the system parameters, so that encryption does not require any pairing computations.

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

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

Let $\vec{a} = H_k(\mathsf{ID}) = a_1 \ldots a_n \in \Sigma^n$. 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 4.2. *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) be a 5-tuple where $g \in \mathbb{G}^*$, $g_1, g_2, g_3 \in \mathbb{G}$, 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 $V = v_1 \ldots v_n \in \Sigma^{(n,m)}$. It then generates an $n \times s$ matrix $U = (u_{i,j})$ as follows. For each $i = 1, \ldots, n$ and $j = 1, \ldots, s$ 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 $\mathit{params} = (g, g_1, g_2, U, k)$. Note that the corresponding (unknown) master secret is $\mathit{master\text{-}key} = g_2^{\alpha}$ 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 $\vec{a} = H_k(\mathsf{ID}) = a_1 \ldots a_n \in \Sigma^n$. If $a_i \neq v_i$ for all $i = 1, \ldots, n$ 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 $r_1, \ldots, r_n \in \mathbb{Z}_p$ and then setting

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

We note that $(d_0, d_1, \ldots, d_n) \in \mathbb{G}^{n+1}$ 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 $(d_0, d_1, \ldots, d_n)$ defined in (2) satisfies

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

where $r_1, \ldots, \tilde{r}_i, \ldots, r_n$ are uniform in $\mathbb{Z}_p$. This matches the definition for a private key for ID and hence $(d_0, d_1, \ldots, d_n)$ 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 $\vec{a} = H_k(\mathsf{ID}^*) = a_1 \ldots a_n \in \Sigma^n$. 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 that $g_3 = g^c$. Then, since $u_{i,a_i} = g^{\alpha_{i,a_i}}$ for all i, we have that

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

Hence, if the tuple (g, g_1, g_2, g_3, T) was sampled from $\mathcal{P}_{\text{BDH}}$, then $T = e(g, g)^{abc} = e(g_1, g_2)^c$ and C is a valid encryption of M_b under ID^*. If on the other hand (g, g_1, g_2, g_3, T) was sampled from $\mathcal{R}_{\text{BDH}}$, then T is random in $\mathbb{G}_1$ and C is independent of b in $\mathcal{A}$'s view.

Phase 2. $\mathcal{A}$ issues more private key queries for identities $\mathsf{ID} \neq \mathsf{ID}^*$, for a total of at most q queries between Phases 1 and 2. The simulator responds as before (aborting as necessary).

Guess. $\mathcal{A}$ outputs a guess $b' \in \{0, 1\}$. The simulator returns 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 selects a random generator $g \in \mathbb{G}^*$, picks a random $\alpha \in \mathbb{Z}_p$, and sets $g_1 = g^\alpha$. Next, it picks a random element $g_2 \in \mathbb{G}$ 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 *params* $= (g, g_1, g_2, U, k)$ and keeps to itself the master secret *master-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 $F(\mathsf{ID}^*) = 0$ then 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 $\mathsf{ID} \neq \mathsf{ID}^*$, for a total of at most q queries between Phases 1 and 2. The simulator responds as before (aborting as necessary).

Guess. $\mathcal{A}$ outputs a guess $b' \in \{0, 1\}$. The simulator returns 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 Appendix A. The proof of Theorem 4.2 will follow immediately from these facts. We define the following notation:

1. Define the random variable $Z = (g, g_1, g_2, g_3, T) \xleftarrow{\text{R}} \mathcal{P}_{\text{BDH}}$.
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 $(F, k) \xleftarrow{\text{R}} \{F_{K,H_k}\}$. 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 $(F, k) \xleftarrow{\text{R}} \{F_{K,H_k}\}$. Then for $b = 0, 1$ we have

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

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 Appendix A. The main theorem follows easily.

Proof of Theorem 4.2. 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

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

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

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

Here, S_j^Z and S_j^Y 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 4.1. We focus on binary alphabets ($s = 2$).

Theorem 5.1. *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 $\gamma \in (0, \frac{1}{2})$.*

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.

Conditional Probability Bounds. 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 $X_i = \{x_1, \ldots, x_i\} = X_i^{\text{neg}} \cup X_i^{\text{pos}}$ where $X_i^{\text{neg}} = \{x \in X_i : F(x) = 0\}$ and $X_i^{\text{pos}} = \{x \in X_i : F(x) = 1\}$, and write $P_j = \Pr[F(x_j) = 1 \mid X_{j-1}^{\text{neg}}, X_{j-1}^{\text{pos}}]$ for the probability we seek to bound. Observe that the two sets X_{j-1}^{neg} and X_{j-1}^{pos} 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

$$\begin{aligned} P_j &= \Pr[F(x_j) = 1 \mid X_{j-1}^{\text{neg}}, X_{j-1}^{\text{pos}}] = \frac{|Y_j \cap S_{j-1}|}{|S_{j-1}|} = \frac{|Y_j \cap S_{j-1}^Y \cap S_{j-1}^Z|}{|S_{j-1}^Y \cap S_{j-1}^Z|} \\ &= \frac{\left|Y_j \cap \left(\bigcap_{x \in X_{j-1}^{\text{pos}}} Y(x)\right) \cap \left(\bigcap_{x \in X_{j-1}^{\text{neg}}} Z(x)\right)\right|}{\left|\left(\bigcap_{x \in X_{j-1}^{\text{pos}}} Y(x)\right) \cap \left(\bigcap_{x \in X_{j-1}^{\text{neg}}} Z(x)\right)\right|} = \frac{\left|Y_j \cap Y_{\cap,j-1}^{\text{pos}} \setminus Y_{\cup,j-1}^{\text{neg}}\right|}{\left|Y_{\cap,j-1}^{\text{pos}} \setminus Y_{\cup,j-1}^{\text{neg}}\right|} \end{aligned}$$

where we have posed $Y_{\cap,j-1}^{\text{pos}} = \left(\cap_{x \in X_{j-1}^{\text{pos}}} Y(x)\right)$ and $Y_{\cup,j-1}^{\text{neg}} = \left(\cup_{x \in X_{j-1}^{\text{neg}}} Y(x)\right)$, or, as expressed in our previous, simpler but less explicit notation, $Y_{\cap,j-1}^{\text{pos}} = S_{j-1}^Y$ and $Y_{\cup,j-1}^{\text{neg}} = \Sigma^{(n,m)} \setminus S_{j-1}^Z$.

In Appendix B.1 we 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:

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

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

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

Detailed calculations for these bounds are given in Appendix B.1.

Statistical Process Discrepancy. 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 construct such an M and bound the probability of discrepancy. Here is the gist of the argument, which we formalize in Appendix B.2.

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 $X_{j-1}^{\text{neg}}, X_{j-1}^{\text{pos}}$, the conditional expectation $P_j = \Pr[F_j = 1 \mid X_{j-1}^{\text{neg}}, X_{j-1}^{\text{pos}}]$ 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, in Appendix B.2 we show that for the process R' to perfectly simulate F, it suffices that for $j = 1, \ldots, q$, the conditional law of $M_j \mid R'_1, \ldots, R'_{j-1}, R_j$ satisfies

$$\begin{aligned} \Pr[M_j = 1, R_j = 0 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] &= \max\{\, 0, \;\; P_j - \delta \,\} \\ \Pr[M_j = 1, R_j = 1 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] &= \max\{\, 0, \;\; \delta - P_j \,\} \end{aligned}$$

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 $\Pr[M_j = 1 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] \leq |P_j - \delta| \leq 1$, which we can bound further using our previous bounds on P_j in the cases where $|X^{\text{pos}}_{j-1}| = 0, 1$. Next, we need to bound the probabilities $\Pr[X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}]$ of the conditioning events. The difficultly here is that the random variables $X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}$ 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 $|X^{\text{pos}}_{j-1}|$. Note that these probabilities vanish quickly as $|X^{\text{pos}}_{j-1}|$ increases, which is why we previously sought bounds for P_j in the cases $|X^{\text{pos}}_{j-1}| = 0, 1$ 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

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

A direct derivation of this inequality may be found in Appendix B.2.

Proof of Theorem 5.1. The theorem now follows easily from the previous bound on the total discrepancy between the PRF oracle $F = F_{K,H}$ and the binomial stochastic process R.

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 $\epsilon_{\text{H}} + \frac{13}{2}\gamma^2/\kappa \leq \epsilon_{\text{PRF}}$, as required. □

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 5.1, given an "ordinary" family of (t, ϵ_{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 5.2. *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 $\mu : \{0,1\}^\beta \to \mathbb{F}^\ell_{2^t}$ 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 $d \times d$ Hadamard matrix is such that any two distinct rows or columns agree on exactly $d/2$ 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

For the sake of concreteness, and to show the feasibility of the construction, we briefly illustrate how to instantiate the various parameters intervening in Theorems 5.1 and 5.2. We assume to be given a bilinear group $\mathbb{G}$ and a hash function family $\bar{\mathcal{H}}$ characterized by:

- β_{H}, the native output length in bits of the collision resistant hash functions;
- ϵ_{H}, the adversarial advantage against the collision resistance assumption on $\bar{\mathcal{H}}$;
- ϵ_{BDH}, the adversarial advantage against the Decision BDH assumption in $\mathbb{G}$.

We are also given q, the maximum number of allowable oracle queries, under the "birthday paradox" guideline that $1 \leq q \ll \sqrt{2^{\beta_{\mathrm{H}}}}$. 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};
2. the time complexity of the four IBE operations is polynomial in the security parameters.

For $s = 2$, we require that $(\epsilon_{\mathrm{IBE}}/\epsilon_{\mathrm{BDH}}) \leq O\Big(\mathrm{poly}(q)\Big)$ and $n \leq O\Big(\mathrm{poly}\big(\beta_{\mathrm{H}}, \log(q), \log(1/\epsilon_{\mathrm{IBE}})\big)\Big)$.

We describe two suboptimal but illustrative settings of the parameters; one favoring security, the other favoring performance. For simplicity, we fix $\kappa \leftarrow 2$ without trying to optimize for κ.

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 bound (defined as $\epsilon'_{\mathrm{PRF}} = \epsilon_{\mathrm{PRF}} - \epsilon_{\mathrm{H}}$ in the

notation of Theorem 5.1) is pegged to $\epsilon'_{\text{PRF}} = \epsilon_{\text{H}}$ or a fraction thereof. We successively assign

$$
\begin{array}{rcl}
\kappa & \leftarrow & 2 \;\; \text{an arbitrary and suboptimal choice to satisfy} \;\; \kappa \in (1,\infty) \\
\gamma & \leftarrow & \sqrt{\epsilon_{\text{H}}}/2 \;\; \text{so that} \;\; \epsilon'_{\text{PRF}} = 13\,\gamma^2/2\,\kappa \lesssim 4\,\gamma^2 = \epsilon_{\text{H}} \;\; \text{in order to meet the designated target} \\
m & \leftarrow & \lceil \log_2(2\,q/\gamma) \rceil \;\; \text{so that} \;\; q \le 2^m\,\gamma/2 = (1-1/s)^{-m}\,\gamma/\kappa \;\; \text{as required by Thm 5.1} \\
\delta & \leftarrow & 2^{-m} \;\; \text{so that} \;\; \delta \lesssim \gamma/2\,q = \gamma/\kappa\,q \;\; \text{with} \;\; \delta = (1-1/s)^m \;\; \text{as required by Thm 5.1} \\
r & \leftarrow & 1/2 - 1/3\,m \;\; \text{an easy target for} \;\; v/n \;\; \text{such that} \;\; 1/2 > r \gtrsim 1 - \sqrt[m]{2}/2 = 1 - \sqrt[m]{\kappa\,\delta} \\
\beta & \leftarrow & \max\{\beta_{\text{H}}, m\} \;\; \text{i.e., the hash output is padded so that} \;\; m \le \beta \le n \;\; \text{by Thm 5.2} \\
t & \leftarrow & \min\{t \in \mathbb{Z}^+ : 2^t \ge \lceil \beta/t \rceil/(1-2\,r) + 1\} \;\; \text{according to the proof of Thm 5.2} \\
\ell & \leftarrow & \lceil \beta/t \rceil \;\; \text{the Reed-Solomon codeword size intervening in the proof of Thm 5.2} \\
n & \leftarrow & 2^t\,(2^t - 1) \;\; \text{the total binary size resulting from the construction of Thm 5.2} \\
v & \leftarrow & 2^{t-1}\,(2^l - \ell) \;\; \text{the Hamming separation achieved by the construction of Thm 5.2} \\
\theta & \leftarrow & (1 - v/n)^m \;\; \text{where} \;\; v/n > r \;\; \text{by Thm 5.2 so that} \;\; \theta < (1-r)^m \le \kappa\,\delta \;\; \text{for Thm 5.1}
\end{array}
$$

Since $t > 1$ and $2^{t-1} - 1 < \lceil \beta/(t-1) \rceil/(1-2\,r) \le 2\,\lceil \beta/t \rceil/(1-2\,r)$ we have $2^t < 6\,m\,\ell + 2$. Substituting in the appropriate expressions, we find—in first rough approximation—that

$$
n \;=\; 2^t\,(2^t - 1) \;<\; (6\,m\,\ell + 2)^2 \;\ll\; (20 + 6\,\log_2(q/\sqrt{\epsilon_{\text{H}}}))^2\,\beta^2 \;=\; O\big(\log_2(q/\sqrt{\epsilon_{\text{H}}}))^2\,\beta^2\big)
$$

Evidently, the total PRF loss $\epsilon_{\text{PRF}} = \epsilon_{\text{H}} + \epsilon'_{\text{PRF}} \lesssim 2\,\epsilon_{\text{H}}$ is negligible. Since m is independent of β_{H}, in the asymptote $\beta = \beta_{\text{H}}$, and thus the bandwidth coefficient $n = O\big(\log_2^2(q/\sqrt{\epsilon_{\text{PRF}}})\,\beta_{\text{H}}^2\big)$ is polynomial in $\log q$ and β_{H}, as required. 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. Suppose for simplicity that the loss ϵ_{H} due to hash collisions is negligible, which in practice also requires that $\beta_{\text{H}} \ge 128$. Under the $(t, \epsilon_{\text{BDH}})$-Decision BDH assumption Theorem 4.2 gives us a $(t, q, \epsilon_{\text{IBE}})$-secure IBE where

$$
\epsilon_{\text{IBE}} \;=\; 2\,\epsilon_{\text{BDH}}/(\delta\,(1-\delta)^q - \epsilon_{\text{PRF}}) \;\approx\; 2\,\epsilon_{\text{BDH}}/(\sqrt{\epsilon_{\text{PRF}}}/4\,q - \epsilon_{\text{PRF}})
$$

We can easily minimize the value of ϵ_{IBE} for a prescribed value of q by seeking $\epsilon_{\text{PRF}} \leftarrow (1/8\,q)^2$. This results in the overall IBE security loss parameter

$$
\epsilon_{\text{IBE}} \;\approx\; 64\,q^2\,\epsilon_{\text{BDH}} \;=\; \Theta\big(q^2\,\epsilon_{\text{BDH}}\big)
$$

For the same arbitrary choice $\kappa \leftarrow 2$ as before, using analogous calculations we find that

$$
\begin{array}{rcl}
\gamma & \leftarrow & 1/16\,q \;\; \text{so that} \;\; \epsilon_{\text{PRF}} \approx \epsilon'_{\text{PRF}} = 13\,\gamma^2/2\,\kappa \lesssim 4\,\gamma^2 = 1/64\,q^2 \\
m & \leftarrow & \lceil \log_2(2\,q/\gamma) \rceil \;=\; \lceil \log_2(32\,q^2) \rceil \;\; \text{so that} \;\; q \le (1-1/s)^{-m}\,\gamma/\kappa \\
\delta & \leftarrow & (1-1/s)^m \;=\; 2^{-m} \;\; \text{so that} \;\; \delta \lesssim 1/32\,q^2 = \gamma/\kappa\,q \\
r & \leftarrow & 1/2 - 1/3\,m \;\; \text{so that} \;\; 1/2 > r \gtrsim 1 - \sqrt[m]{2}/2 = 1 - \sqrt[m]{\kappa\,\delta} \\
\beta & \leftarrow & \max\{\beta_{\text{H}}, m\} \;\; \text{padding the hash so that} \;\; m \le \beta \le n \\
t & \leftarrow & \min\{t \in \mathbb{Z}^+ : 2^t \ge \lceil \beta/t \rceil/(1-2\,r) + 1\} \\
\ell & \leftarrow & \lceil \beta/t \rceil\,, \qquad n \;\leftarrow\; 2^t\,(2^t - 1)\,, \qquad v \;\leftarrow\; 2^{t-1}\,(2^l - \ell) \\
\theta & \leftarrow & (1 - v/n)^m \;\; \text{so that} \;\; \theta < (1-r)^m \le \kappa\,\delta
\end{array}
$$

and thus, for $\beta \geq 128$ and non-zero q, we successively bound

$$m \;\geq\; 5\,, \qquad r \;\geq\; 13/30\,, \qquad t \geq 8\,, \qquad 2^t < 1.026 + 0.385\, m\, \beta$$

$$n \;<\; (1.026 + 0.385\, m\, \beta)^2 \;\ll\; (3.34 + 0.77\, \log_2 q)^2\, \beta^2 \;=\; O\big((\log_2 q)^2\, \beta^2\big)$$

The bandwidth coefficient $n = O((\log_2 q)^2\, \beta_{\text{H}}^2)$ remains large, but is an improvement over the previous case.

We note that the optimal value of κ varies and is tied to the particular 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. Hierarchical identities were introduced by Horwitz and Lynn [HL02], and a Hierarchical IBE (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 (IND-ID-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 [BB04a]. The details are deferred to the full paper.

Chosen Ciphertext Security. A recent result of Canetti et al. [CHK04] gives an efficient way to build a chosen ciphertext IBE (IND-ID-CCA) from a chosen plaintext 2-HIBE (IND-ID-CPA). Thus, by the previous paragraph, we obtain a full chosen identity, chosen ciphertext IBE (IND-ID-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 IND-ID-CPA secure then so is the scheme with the additional hash. This holds for the HIBE and the chosen ciphertext secure system 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 with a polynomial time security reduction 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 with a tight security reduction without random oracles, based on Decision BDH or a comparable assumption.

References

[BB04a] Dan Boneh and Xavier Boyen. Efficient selective-ID identity based encryption without random oracles. In *Advances in Cryptology—EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 223–238. Springer-Verlag, 2004.

[BB04b] Dan Boneh and Xavier Boyen. Efficient selective-ID identity based encryption without random oracles. Full version of [BB04a], 2004. Available online at: `http://www.cs.stanford.edu/~xb/eurocrypt04b/`.

[BBP04] Mihir Bellare, Alexandra Boldyreva, and Adriana Palacio. An uninstantiable random-oracle-model scheme for a hybrid-encryption problem. In *Advances in Cryptology—EUROCRYPT 2004*, 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 2001*, volume 2139 of *LNCS*, pages 213–229. Springer-Verlag, 2001.

[BF03] Dan Boneh and Matt Franklin. Identity-based encryption from the Weil pairing. *SIAM Journal 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 2003*, volume 2729 of *LNCS*, pages 383–399. Springer-Verlag, 2003.

[BR93] Mihir Bellare and Phil Rogaway. Random oracle are practical: A paradigm for designing efficient protocols. In *ACM Conference on Computer and Communications Security—CCS 1993*, 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 2003*, volume 2567 of *LNCS*. Springer-Verlag, 2003.

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

[CHK03] Ran Canetti, Shai Halevi, and Jonathan Katz. A forward-secure public-key encryption scheme. In *Advances in Cryptology—EUROCRYPT 2003*, 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 2004*, 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 2002*, 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 2004*, volume 2964 of *LNCS*, pages 67–80, 2004.

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

[Jou00] Antoine Joux. A one round protocol for tripartite Diffie-Hellman. In *Algorithmic Number Theory Symposium—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 1984*, volume 196 of *LNCS*, pages 47–53. Springer-Verlag, 1984.

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

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

[TI89] Shigeo Tsujii and Toshiya Itoh. An id-based cryptosystem based on the discrete logarithm problem. *IEEE Journal on Selected Areas in Communication*, 7(4):467–473, 1989.

A Proving Theorem 4.2

We now prove the four claims used in Section 4.3 to establish Theorem 4.2.

Proof of Claim 1

Proof. The IBE system of Section 4.2 essentially instantiates the selective-ID secure, Decision BDH based, Hierarchical IBE (HIBE) of Boneh and Boyen [BB04a, BB04b, §4] into an n-level HIBE, whose k-th level now corresponds to the k-th symbol of the hashed identity string.

By the same argument as in the Boneh-Boyen HIBE proof of security, it can be shown that when $(g, g_1, g_2, g_3, T) \xleftarrow{\text{R}} \mathcal{P}_{\text{BDH}}$ the simulation provided by Experiment 1 is perfect when it does not abort. (The main difference between the present simulator and that of the Boneh-Boyen HIBE is that here the "hierarchy" of private key components is "reshuffled" on the fly depending on the available components; this does not affect the applicability of the simulation argument.)

Since by design Experiment 2 aborts with the prescribed probability and is perfect when it does not abort, it follows that the system parameters in both experiments are generated from the same distribution. Similarly, the responses to all private key queries as well as the challenge ciphertext in both experiments are generated from the same distribution. Therefore, $\mathcal{A}$'s output is both experiments is sampled from the same distribution. □

Proof of Claim 2

Proof. Let $(F, k) \xleftarrow{\text{R}} \{F_{K,H_k}\}$. By Claim 1 it suffices to show that $\Pr[\mathsf{CPA\text{-}Exp}_{\mathcal{A}}(b) = 1]$ is equal to $\Pr[\mathsf{PRF\text{-}Exp}_{\mathcal{A}}(b, F, k) = 1 \mid \mathsf{PRF\text{-}Exp}_{\mathcal{A}}(b, F, k) \neq \mathsf{abort}]$. Observe that experiment $\mathsf{PRF\text{-}Exp}_{\mathcal{A}}(b, F, k)$ is identical to $\mathsf{CPA\text{-}Exp}_{\mathcal{A}}(b)$ except that we add an artificial abort condition before responding to private key queries and before generating the challenge ciphertext. If the abort condition never happens, then, from $\mathcal{A}$'s view, the two experiments are identical. The claim now follows. □

Proof of Claim 3

Proof. Let $\delta = (1 - 1/s)^m$ and let $F_r \xleftarrow{\text{R}} \Omega_\delta$ be a random function with bias δ. Let $k_r \xleftarrow{\text{R}} \mathcal{K}$. Then, it easy to see that

$$\Pr[\mathsf{PRF\text{-}Exp}_{\mathcal{A}}(b, F_r, k_r) = \mathsf{abort}] = 1 - \delta(1 - \delta)^q = 1 - \Delta$$

Recall that $\{H_k : \{0,1\}^w \to \Sigma^n\}_{k \in \mathcal{K}}$ is a $(t, \epsilon_{\text{PRF}}, q, m)$-admissible family of hash functions. It now follows that for $(F, k) \xleftarrow{\text{R}} \{F_{K,H_k}\}$ we have

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

as required. □

Proof of Claim 4

Proof. We construct an algorithm that has advantage at least $(\Delta - \epsilon_{\text{PRF}})|t_0 - t_1|/2$ in distinguishing a 5-tuple (g, g_1, g_2, g_3, T) drawn from $\mathcal{P}_{\text{BDH}}$ from a 5-tuple drawn from $\mathcal{R}_{\text{BDH}}$. This will prove that $(\Delta - \epsilon_{\text{PRF}})|t_0 - t_1|/2$ must be less than ϵ_{BDH} as required.

On input $Z = (g, g_1, g_2, g_3, T)$ the distinguishing algorithm works as follows:

1. Pick a random $b \in \{0, 1\}$.

2. Run experiment $\mathsf{BDH\text{-}Exp}_{\mathcal{A}}(b,(g,g_1,g_2,g_3,T))$. Denote the output by $\mathrm{Out}_{\mathcal{A}} \in \{0,1,\mathsf{abort}\}$.
3. If $\mathrm{Out}_{\mathcal{A}} = \mathsf{abort}$ then output a random $c \in \{0,1\}$ and stop.
4. Otherwise, $\mathrm{Out}_{\mathcal{A}} \in \{0,1\}$. if $\mathrm{Out}_{\mathcal{A}} = b$ output 0, else output 1.

Denote the output of this algorithm by $\mathcal{B}(Z)$. First, when Z is sampled from $\mathcal{P}_{\mathrm{BDH}}$ we have

$$\begin{aligned}
\Pr[\mathcal{B}(Z)=0] &= \Pr[\mathcal{B}(Z)=0 \mid \mathrm{Out}_{\mathcal{A}} \neq \mathsf{abort}] \cdot \Pr[\mathrm{Out}_{\mathcal{A}} \neq \mathsf{abort}] + \\
&\qquad \Pr[\mathcal{B}(Z)=0 \mid \mathrm{Out}_{\mathcal{A}} = \mathsf{abort}] \cdot \Pr[\mathrm{Out}_{\mathcal{A}} = \mathsf{abort}] \\
&= \Pr[\mathrm{Out}_{\mathcal{A}} = b \mid \mathrm{Out}_{\mathcal{A}} \neq \mathsf{abort}] \cdot \Pr[\mathrm{Out}_{\mathcal{A}} \neq \mathsf{abort}] + \frac{1}{2}\Pr[\mathrm{Out}_{\mathcal{A}} = \mathsf{abort}] \\
&= \frac{1}{2}(1-t_0+t_1)\Pr[\mathrm{Out}_{\mathcal{A}} \neq \mathsf{abort}] + \frac{1}{2}\Pr[\mathrm{Out}_{\mathcal{A}} = \mathsf{abort}] \\
&= \frac{1}{2} + \frac{1}{2}(t_1-t_0)\Pr[\mathrm{Out}_{\mathcal{A}} \neq \mathsf{abort}]
\end{aligned}$$

Next, observe that $\Pr[\mathcal{B}(Z)=0 \ : \ Z \xleftarrow{\mathrm{R}} \mathcal{R}_{\mathrm{BDH}}] = 1/2$. Indeed, when $Z \xleftarrow{\mathrm{R}} \mathcal{R}_{\mathrm{BDH}}$ we have that the bit b used to create the challenge ciphertext in the experiment is independent of $\mathcal{A}$'s view. Therefore, $\Pr[\mathrm{Out}_{\mathcal{A}} = b \mid \mathrm{Out}_{\mathcal{A}} \neq \mathsf{abort}] = 1/2$. As above, it follows that $\Pr[\mathcal{B}(Z)=0 \ : \ Z \xleftarrow{\mathrm{R}} \mathcal{R}_{\mathrm{BDH}}] = 1/2$.

Putting these equalities together, we obtain

$$\begin{aligned}
\epsilon_{\mathrm{BDH}} &> \left|\Pr\left[\mathcal{B}(Z)=0 \ : \ Z \xleftarrow{\mathrm{R}} \mathcal{P}_{\mathrm{BDH}}\right] - \Pr\left[\mathcal{B}(Z)=0 \ : \ Z \xleftarrow{\mathrm{R}} \mathcal{R}_{\mathrm{BDH}}\right]\right| \\
&= \frac{1}{2}|t_0-t_1| \ \Pr\left[\mathrm{Out}_{\mathcal{A}} \neq \mathsf{abort} \ : \ Z \xleftarrow{\mathrm{R}} \mathcal{P}_{\mathrm{BDH}}\right] \ > \ (\Delta - \epsilon_{\mathrm{PRF}})|t_0-t_1|/2
\end{aligned}$$

where the last inequality follows from Claims 1 and 3. □

B Proving Theorem 5.1

We now establish the various bounds intervening in the proof of Theorem 5.1. We start by showing the following general inequality in the setting of Section 5.1, which will serve us in Section B.1.

Lemma B.1. *In the conditions of Theorem 5.1,* $|Y_i \cap Y_j| \ \leq \ \theta\,|Y_i|$.

Proof. In virtue of the Hamming separation property, we know that for any x_i, x_j, the corresponding hashes h_i, h_j will disagree at a minimum of v coordinates. Suppose that h_i, h_j disagree at coordinate ℓ. Then for each $K \in Y_i \cap Y_j$ such that $K|_\ell \neq \perp$, it must be the case that $K|_\ell \notin \{h_i|_\ell, h_j|_\ell\}$. Thus, for each such K we have eliminated two possible choices for the value of $K|_\ell$. If on the other hand h_i, h_j agree at coordinate ℓ, then we know that $K|_\ell \notin \{h_i|_\ell\}$, eliminating only one of the possible values for $K|_\ell$. Carrying out this reasoning for all $\binom{n}{m}$ possible choices for the support of K, we deduce that

$$|Y_i \cap Y_j| \ \leq \ \sum_{i=\max\{0,m+v-n\}}^{\min\{m,v\}} (s-1)^{m-i}(s-2)^i \binom{n-v}{m-i}\binom{v}{i} \ = \ \bar{\theta}\,|Y_i|$$

where we have defined $\bar{\theta} = \sum_{i=\max\{0,m+v-n\}}^{\min\{m,v\}} \left(\frac{s-2}{s-1}\right)^i \binom{n-v}{m-i}\binom{v}{i}/\binom{n}{m}$. In the case of a binary alphabet, the above simplifies greatly to yield

$$|Y_i \cap Y_j| \ \leq \ \binom{n-v}{m} \cdot \left\{ \begin{array}{ll} 1 & \text{if } m \leq n-v \\ 0 & \text{otherwise} \end{array} \right\} \ \leq \ \binom{n}{m}(1-v/n)^m \ = \ \theta\,|Y_i| \qquad (\text{for } s=2)$$

where $\theta = (1-v/n)^m$ as in Theorem 5.1. □

B.1 Probability Bounds

We now derive the uniform bounds on the conditional probabilities $P_j = \Pr[F(x_j) = 1 \mid X_{j-1}^{\text{neg}}, X_{j-1}^{\text{pos}}]$ in function of $|X_{j-1}^{\text{neg}}|$, as stated in Section 5.1.

Claim: $P_j \leq (1+2\gamma)\delta$ **for all** $X_{j-1}^{\text{neg}}, X_{j-1}^{\text{pos}}$ **such that** $|X_{j-1}^{\text{pos}}| = 0$

Proof. Since $X_{j-1}^{\text{pos}} = \emptyset$, by a simple counting argument that only depends on the sizes of the alphabet and the hash vectors, we obtain

$$\begin{aligned} P_j &= \left|Y_j \setminus Y_{\cup,j-1}^{\text{neg}}\right| \;/\; \left|\Sigma^n \setminus Y_{\cup,j-1}^{\text{neg}}\right| \;\leq\; |Y_j| \;/\; \left(|\Sigma^n| - \sum_{i=1}^{j-1} |Y_i|\right) \\ &\leq \frac{\binom{n}{m}(s-1)^m}{\binom{n}{m}s^m - (j-1)\cdot\binom{n}{m}(s-1)^m} \;\leq\; \delta/(1-q\,\delta) \;\leq\; (1-\gamma)^{-1}\,\delta \;\leq\; (1+2\gamma)\,\delta \end{aligned}$$

where the last two inequalities stems from the constraints $q \leq \gamma/\kappa\,\delta \leq \gamma/\delta$ and $\gamma < 1/2$. □

Claim: $P_j \geq (1-\gamma)\delta$ **for all** $X_{j-1}^{\text{neg}}, X_{j-1}^{\text{pos}}$ **such that** $|X_{j-1}^{\text{pos}}| = 0$

Proof. Since $X_{j-1}^{\text{pos}} = \emptyset$, we appeal to the following counting argument, using Lemma B.1, to obtain

$$\begin{aligned} P_j &= \left|Y_j \setminus Y_{\cup,j-1}^{\text{neg}}\right| \;/\; \left|\Sigma^n \setminus Y_{\cup,j-1}^{\text{neg}}\right| \;\geq\; \left(|Y_j| - \sum_{i=1}^{j-1} |Y_j \cap Y_i|\right) \;/\; |\Sigma^n| \\ &\geq (1-(j-1)\cdot\theta)\,\delta \;\geq\; (1-q\cdot\theta)\,\delta \;\geq\; (1-\gamma)\,\delta \end{aligned}$$

where the last inequality stems from the constraint $q \leq \gamma/\kappa\,\delta \leq \gamma/\theta$, in the case $s = 2$. □

Claim: $P_j \leq 2\kappa\delta$ **for all** $X_{j-1}^{\text{neg}}, X_{j-1}^{\text{pos}}$ **such that** $|X_{j-1}^{\text{pos}}| = 1$

Proof. We can assume without loss of generality that $X_{j-1}^{\text{pos}} = \{x_{i_*}\}$ and $X_{j-1}^{\text{neg}} = \{x_{i_1}, \ldots, x_{i_{j-2}}\}$. We modify our counting argument accordingly, again using Lemma B.1, to obtain

$$\begin{aligned} P_j &= \left|Y_j \cap Y_{i_*} \setminus Y_{\cup,j-1}^{\text{neg}}\right| \;/\; \left|Y_{i_*} \setminus Y_{\cup,j-1}^{\text{neg}}\right| \;\leq\; |Y_j \cap Y_{i_*}| \;/\; \left(|Y_{i_*}| - \sum_{i=1}^{j-2} |Y_{i_*} \cap Y_{i_i}|\right) \\ &\leq \frac{\theta}{1-(j-2)\cdot\theta} \;\leq\; \frac{\theta}{1-q\cdot\theta} \;\leq\; \frac{\kappa\,\delta}{1-q\cdot\kappa\,\delta} \;\leq\; \kappa(1-\gamma)^{-1}\,\delta \;\leq\; 2\,\kappa\,\delta \end{aligned}$$

where the last three steps stem from the constraints $\theta \leq \kappa\,\delta$ and $q \leq \gamma/\kappa\,\delta \leq (2\,\kappa\,\delta)^{-1}$, for $s = 2$. □

B.2 Process Discrepancy

We now bound the statistical distance $D(\langle F_1, \ldots, F_q\rangle, \langle R_1, \ldots, R_q\rangle)$ between an interactively sampled q-prefix of the bias map oracle F with uniform random key $K \in \Sigma^{(n,m)}$ and a q-prefix of the binomial stochastic process oracle R from Section 5.1.

Recall that the statistical distance between two random variables A and B taking values in the same discrete domain Ω is the quantity $D(A,B) \in [0,1]$ given by

$$D(A,B) = \frac{1}{2}\sum_{\omega\in\Omega} \big|\Pr[A=\omega] - \Pr[B=\omega]\big|$$

Hence, $D(A, B)$ is the fraction of the distributions of A and B that disagree with each other, and is thus the maximum probability or advantage with which the two distributions can be distinguished from two respective samples $a \xleftarrow{\text{R}} A$ and $b \xleftarrow{\text{R}} B$. In other words, for any algorithm $\mathcal{A}$ we have

$$\left| \Pr[\mathcal{A}(a) = 1 : a \xleftarrow{\text{R}} A] - \Pr[\mathcal{A}(b) = 1 : b \xleftarrow{\text{R}} B] \right| \leq D(A, B)$$

We show that $D(\langle F_1, \ldots, F_q \rangle, \langle R_1, \ldots, R_q \rangle) \leq \frac{13}{2}\gamma^2/\kappa$ where $F = F_{H,K}$ for random $K \in \Sigma^{(n,m)}$ when the samples $F_1, \ldots, F_q$ can be queried adaptively.

Let thus R be the reference binomial process, M the control process, R' the simulated process, and $F = F_{H,K} : \{0,1\}^w \to \{0,1\}$ the bias map oracle with public hash function H and secret key K. Recall that E_j for any $j \leq q$ denotes the event $[\exists i \leq j : M_i = 1]$. We specify M so that R' perfectly simulates F for some $K \in \Sigma^{(n,m)}$ chosen uniformly at random (albeit unknown to the simulator). The simulation shows that the distributions of $\langle F_1, \ldots, F_q \rangle$ and $\langle R_1, \ldots, R_q \rangle$ have at least $\Pr[\neg E_q]$ probability mass in common, so that necessarily $D(\langle F_1, \ldots, F_q \rangle, \langle R_1, \ldots, R_q \rangle) \leq \Pr[E_q]$. We then find an upper bound on the probability of E_q for our specific construction, which gives us the desired result.

For $j = 0, \ldots, q$, we define the random variables (or statistics) $\Sigma R_j = \sum_{i=1}^{j} R_i$, $\Sigma R'_j = \sum_{i=1}^{j} R'_i$, and $\Sigma M_j = \sum_{i=1}^{j} M_i$. Notice that E_j and $[\Sigma M_j \neq 0]$ denote the same event. Since the adversary interacts with a simulated oracle R' instead of the actual oracle F, the adversary's information is determined by the simulated outcomes $(R')_{1,\ldots,q}$. Specifically, the knowledge state before step j is captured by the partition $\bar{X}^{\text{neg}}_{j-1} \cup \bar{X}^{\text{pos}}_{j-1} = X_{j-1}$ of the query history $X_{j-1} = \{x_i : i = 1, \ldots, j-1\}$, where membership of x_i to either $\bar{X}^{\text{neg}}_{j-1}$ or $\bar{X}^{\text{pos}}_{j-1}$ depends on the value of R'_i. In particular, we have that $|\bar{X}^{\text{pos}}_j| = \Sigma R'_j$. In the adversary's view, the conditional expectation of the process at step j of the interaction is given by $P'_j = \Pr[R'_j = 1 \mid \bar{X}^{\text{neg}}_{j-1}, \bar{X}^{\text{pos}}_{j-1}]$. Thus, the simulation is perfect if $P'_j = P_j$ given the same knowledge state, i.e., if we have $\Pr[R'_j = 1 \mid \bar{X}^{\text{neg}}_{j-1}, \bar{X}^{\text{pos}}_{j-1}] = \Pr[F_j = 1 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}]$ whenever $\langle \bar{X}^{\text{neg}}_{j-1}, \bar{X}^{\text{pos}}_{j-1} \rangle = \langle X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1} \rangle$ for $j = 1, \ldots, q$. For clarity, in the sequel we drop the notation $\bar{X}^{\text{neg}}_{j-1}, \bar{X}^{\text{pos}}_{j-1}$ and rewrite the adversary's knowledge state in the simulation as $X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}$. We assume the values of P_j and P'_j to be defined in reference to that knowledge state.

Claim: $(R')_{1,\ldots,q}$ **simulates** $(F_{H,K})_{1,\ldots,q}$ **for uniform random** $K \in \Sigma^{(n,m)}$

Proof. We construct M by specifying the conditional law of M_j given $X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}, R_j$, which we express as follows

$$\begin{aligned} \Pr[M_j = 1, R_j = 0 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] &= \max\{\ 0,\ \ P_j - \delta\ \} \\ \Pr[M_j = 1, R_j = 1 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] &= \max\{\ 0,\ \ \delta - P_j\ \} \end{aligned}$$

We need to show that this specification causes R' to be distributed as $F = F_{H,K}$ for uniform random secret K in the adversary's view. First, we note that the conditional law of $M_j \mid R'_1, \ldots, R'_{j-1}, R_j$ is well defined. Next, we show by induction that it leads to the correct distribution. Initially, the distribution of K in the adversary's view is uniform over $S_0 = \Sigma^{(n,m)}$. Now, assume that after the completion of step $j-1$ the distribution of K in the adversary's view is uniform over the set S_{j-1} (where S_{j-1} is defined given the query history represented by $X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}$ as in Section 5). The expectation of F_j conditionally on the adversary's knowledge state is thus given by P_j. By

construction of M_j, it is easy to see that R'_j has the same conditional expectation, to wit

$$\begin{aligned} P'_j &= \Pr[R_j = 1, M_j = 0 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] \\ &\quad + \Pr[R_j = 0, M_j = 1 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] \\ &= \Pr[R_j = 1 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] - \Pr[R_j = 1, M_j = 1 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] \\ &\quad + \Pr[R_j = 0, M_j = 1 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] \\ &= \delta - \max\{0,\ \delta - P_j\} + \max\{0,\ P_j - \delta\} \ = \ P_j \end{aligned}$$

It follows that $R'_j \mid R'_1, \ldots, R'_{j-1}$ has the same conditional expectation as an oracle $F = F_{H,K}$ for uniform random $K \in S_{j-1}$. Therefore, the adversary gains the same information in either scenario, and thus, after R'_j is revealed, K appears uniformly distributed in the subset $S_j \subseteq S_{j-1}$. By induction, we conclude that the successive outcomes R'_j for $j = 1, \ldots, q$ have the same conditional expectations as $F_{H,K_{j-1}}$ for uniform random $K_{j-1} \in S_{j-1}$, where $S_{q-1} \subseteq \ldots \subseteq S_1 \subseteq S_0 = \Sigma^{(n,m)}$. Consequently, in the adversary's view, the sequence $\langle R'_1, \ldots, R'_q \rangle$ from the interactive simulation is distributed as a sequence $\langle F_1, \ldots, F_q \rangle$ from an oracle $F = F_{H,K}$ for some unknown K initially uniform in $\Sigma^{(n,m)}$, as required. □

Claim: $\Pr[E_q] \leq \frac{13}{2}\gamma^2/\kappa$ **for this construction of** M **and** R'

Proof. It remains to bound the unconditional probability of the event E_q. Unfortunately, $\Pr[E_q]$ depends on the law of M, which is problematic in two respects. First, it depends on the adversary's knowledge state; second, it is function of P_j which is difficult to compute even given the query history.

However, for any k such that there exists $\underline{P}^{(k)}_j$ and $\overline{P}^{(k)}_j$ for which it holds that $\underline{P}^{(k)}_j \leq P_j \leq \overline{P}^{(k)}_j$ uniformly over all histories $X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}$ satisfying $\Sigma R'_{j-1} = k$, we can write

$$\begin{aligned} \forall X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1} \ \text{ s.t. } \ \Sigma R'_{j-1} = k \ : \quad \Pr[M_j = 1, R_j = 0 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] &\leq \max\{\ 0,\ \ \overline{P}^{(k)}_j - \delta\ \} \\ \forall X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1} \ \text{ s.t. } \ \Sigma R'_{j-1} = k \ : \quad \Pr[M_j = 1, R_j = 1 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] &\leq \max\{\ 0,\ \ \delta - \underline{P}^{(k)}_j\ \} \end{aligned}$$

On the one hand, in particular, using our previously computed bounds for $k = 0$, we already know

$$\forall X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1} \ \text{ s.t. } \ \Sigma R'_{j-1} = 0 \ : \ \begin{cases} \Pr[M_j = 1, R_j = 0 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] \ \leq \ 2\gamma\delta \\ \Pr[M_j = 1, R_j = 1 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] \ \leq \ \gamma\delta \end{cases}$$

and thus, as any finite convex combination is enclosed in the convex hull of the combinants,

$$\begin{aligned} \Pr[M_j = 1, R_j = 0 \mid \Sigma R'_{j-1} = 0] &\leq 2\gamma\delta \\ \Pr[M_j = 1, R_j = 1 \mid \Sigma R'_{j-1} = 0] &\leq \gamma\delta \end{aligned}$$

Also, using our upper bound for $k = 1$ and the fact that $\Pr[R_j = 1 \mid \text{AnyEvent}_{time<j}] = \delta$, we have

$$\forall X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1} \ \text{ s.t. } \ \Sigma R'_{j-1} = 1 \ : \ \begin{cases} \Pr[M_j = 1, R_j = 0 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] \ \leq \ 2\kappa\delta \\ \Pr[M_j = 1, R_j = 1 \mid X^{\text{neg}}_{j-1}, X^{\text{pos}}_{j-1}] \ \leq \ \delta \end{cases}$$

and thus, again using the fact that a finite convex combination lies within the convex hull,

$$\begin{aligned} \Pr[M_j = 1, R_j = 0 \mid \Sigma R'_{j-1} = 1] &\leq 2\kappa\delta \\ \Pr[M_j = 1, R_j = 1 \mid \Sigma R'_{j-1} = 1] &\leq \delta \end{aligned}$$

On the other hand, since when $\Sigma M_i = 0$ the sequence $\langle R'_1, \ldots, R'_i \rangle$ identifies with $\langle R_1, \ldots, R_i \rangle$, it follows that for any suitable predicate $f : \{0,1\}^{j-1} \to \{\bot, \top\}$ we have

$$\Pr[M_j = 1,\ \Sigma M_{j-1} = 0 \mid f(R_1, \ldots, R_{j-1})] \ = \ \Pr[M_j = 1,\ \Sigma M_{j-1} = 0 \mid f(R'_1, \ldots, R'_{j-1})] \qquad (3)$$

Therefore, by decomposing the probability of interest $\Pr[E_q]$ over disjoint events, and manipulating the summands using the preceding results, we easily find that

$$\begin{aligned}
\Pr[E_q] \ &= \ \Pr[\Sigma M_q \neq 0] \ = \ \sum_{j=1}^{q} (\Pr[M_j = 1, \Sigma M_{j-1} = 0]) \\
&= \ \sum_{j=1}^{q} \sum_{r=0}^{1} \Pr[M_j = 1, \Sigma M_{j-1} = 0, \Sigma R_{j-1} = r] + \sum_{j=1}^{q} \sum_{r=2}^{q-1} \Pr[M_j = 1, \Sigma M_{j-1} = 0, \Sigma R_{j-1} = r] \\
&\leq \ \left(\sum_{j=1}^{q} \sum_{r=0}^{1} \Pr[M_j = 1, \Sigma M_{j-1} = 0 \mid \Sigma R_{j-1} = r] \cdot \Pr[\Sigma R_{j-1} = r] \right) + \Pr[\Sigma R_q \geq 2] \\
&\overset{(\star)}{=} \ \left(\sum_{j=1}^{q} \sum_{r=0}^{1} \Pr[M_j = 1, \Sigma M_{j-1} = 0 \mid \Sigma R'_{j-1} = r] \cdot \Pr[\Sigma R_{j-1} = r] \right) + \Pr[\Sigma R_q \geq 2] \\
&\leq \ \left(\sum_{j=1}^{q} \sum_{r=0}^{1} \left(\begin{array}{c} \Pr[M_j = 1, R_j = 0 \mid \Sigma R'_{j-1} = r] \\ + \Pr[M_j = 1, R_j = 1 \mid \Sigma R'_{j-1} = r] \end{array} \right) \cdot \Pr[\Sigma R_q \geq r] \right) + \Pr[\Sigma R_q \geq 2] \\
&\leq \ q(2\gamma\delta + \gamma\delta) + q(2\kappa\delta + \delta)q\delta + \frac{1}{2}q^2\delta^2 \ \leq \ 3\gamma^2/\kappa + \gamma^2(2/\kappa + 1/\kappa^2) + \frac{1}{2}\gamma^2/\kappa^2 \ \leq \ \frac{13}{2}\gamma^2/\kappa
\end{aligned}$$

where at step $(\star)$ we have used (3) with the predicate $f(r_1, \ldots, r_{j-1})$ instantiated as $\sum_{i=1}^{j-1} r_i \overset{?}{=} r$. We conclude that $D(\langle F_1, \ldots, F_q \rangle, \langle R_1, \ldots, R_q \rangle) \leq \frac{13}{2}\gamma^2/\kappa$, as required. $\square$