On Notions of Security for Deterministic Encryption, and Efficient Constructions Without Random Oracles

Alexandra Boldyreva[1], Serge Fehr[2] and Adam O'Neill[1]

[1] Georgia Institute of Technology, Atlanta, GA, USA
{aboldyre,amoneill}@cc.gatech.edu
[2] CWI, Amsterdam, Netherlands
Serge.Fehr@cwi.nl

Abstract. The study of deterministic public-key encryption was initiated by Bellare et al. (CRYPTO '07), who provided the "strongest possible" notion of security for this primitive (called PRIV) and constructions in the random oracle (RO) model. We focus on constructing efficient deterministic encryption schemes *without* random oracles. To do so, we propose a slightly weaker notion of security, saying that no partial information about encrypted messages should be leaked as long as each message is a-priori hard-to-guess *given the others* (while PRIV did not have the latter restriction). Nevertheless, we argue that this version seems adequate for many practical applications. We show equivalence of this definition to single-message and indistinguishability-based ones, which are easier to work with. Then we give general constructions of both chosen-plaintext (CPA) and chosen-ciphertext-attack (CCA) secure deterministic encryption schemes, as well as efficient instantiations of them under standard number-theoretic assumptions. Our constructions build on the recently-introduced framework of Peikert and Waters (STOC '08) for constructing CCA-secure *probabilistic* encryption schemes, extending it to the deterministic-encryption setting as well.

1 Introduction

1.1 Background and Overview

Motivation. Deterministic public-key encryption (where the encryption algorithm is deterministic) was studied by Bellare, Boldyreva and O'Neill [1]. They proposed a semantic-security-style definition of privacy for it, called PRIV, which requires that no partial information about multiple, possibly-dependent messages is leaked from their encryptions, while appropriately taking into account two inherent limitations of deterministic encryption: privacy is only possible for messages that are a-priori hard-to-guess by the adversary, and some information about a message leaks unavoidably, namely its encryption. Both the chosen-plaintext (CPA) and chosen-ciphertext-attack (CCA) cases were considered, and the authors designed several constructions meeting them.

Deterministic encryption seems interesting and useful. As discussed in [1], it allows for fast searching on encrypted data; moreover, deterministic encryption can be length-preserving, which can be needed for securing legacy code or in bandwidth-critical applications. Finally, we find that the study of deterministic encryption can have applications to normal (randomized) encryption as well.

However, the constructions of [1] are only proven secure in the random oracle (RO) model [4]. Of course, finding alternative schemes secure in the standard model (i.e. without random oracles) is desirable, as a growing number of papers have raised concerns about the "soundness" of the RO model (e.g. [8, 22, 2] to name a few). Finding deterministic encryption schemes secure in the standard model was left as an important open problem in [1].

This paper. We construct efficient deterministic encryption schemes without random oracles, secure under standard number-theoretic assumptions. The notion of security we use, however, is slightly weaker than that of [1], in that it considers the encryption of *block-sources*. That is, it guarantees no partial information about encrypted messages is leaked, as long as each message is a-priori hard-to-guess *given the other messages*. We believe this notion to nevertheless be suitable for a variety of practical applications, for example the encryption high-entropy data containing social security or phone numbers. In such examples, messages can depend on one another, e.g. share a common prefix, yet the foregoing condition is satisfied.

Related work. The encryption of high-entropy sources was first considered in the information-theoretic, symmetric-key setting by Russell and Wang [28], and the problem was studied in greater generality (under the name "entropic security") by Dodis and Smith [20, 19]. Entropic security was later studied in the quantum setting by Desrosiers and Dupuis [16, 17].

1.2 Main Results

Equivalent Definitions. We show that PRIV-security for block-sources is equivalent to PRIV-security for a *single* hard-to-guess message. The latter was briefly introduced (using a slightly different formulation) in [1] under the name PRIV1, where it was shown *strictly weaker* than PRIV, but beyond that this notion remained unstudied. We also show equivalence of PRIV1 to a single-message, indistinguishability-based notion, which is handier to work with. The proof is non-trivial and employs ideas from [20] and [16, 17], used for showing the equivalence between entropic security for information-theoretic symmetric-key (quantum) encryption schemes and an indistinguishability-based notion. All our results about the definitions extend to the CCA setting as well.

General constructions. We present general constructions of both CPA- and CCA-secure deterministic encryption schemes, building on the recently-introduced framework of Peikert and Waters [26] for constructing (randomized) IND-CCA encryption schemes in the standard model. Recall that [26] introduces a framework of "lossy" trapdoor functions (TDFs) — TDFs that operate in one two possible "modes," an injective one and an un-invertible lossy one, for which the outputs are indistinguishable. We observe that if the lossy mode also acts as a *universal hash function* [9, 10] (in which case we say it has a *universal hash mode*), then the lossy TDF in injective mode is in fact a secure deterministic encryption scheme in our sense. Indeed, this follows straightforwardly under our indistinguishability-based security notion by the Leftover-Hash Lemma (LHL) [23, 6]. We extend the connection between lossy TDFs and deterministic encryption schemes to the CCA setting as well: our general CCA-secure construction can be viewed as a "deterministic" version of the general IND-CCA scheme of [26]. Unlike the latter it does not use a one-time signature scheme but rather a hash function H that is both target-collision resistant (TCR) [24, 5] and universal. It also uses a lossy TDF F and an all-but-one (ABO) TDF G (the latter is a generalization of the former introduced in [26] whose first input is drawn from a set of *branches*, one of which is lossy), where as before lossiness must be strengthened to universality. The encryption of message m under our scheme has the form $(H(m), F(m), G(H(m), m))$.

DDH-BASED INSTANTIATIONS. We obtain instantiations of our general constructions based on the decisional Diffie-Hellman assumption (DDH) rather straightforwardly. In fact, we show that the DDH-based lossy and ABO TDF constructs of [26] already suffice; that is, they indeed have "universal" lossy modes. To construct an appropriate hash function for our CCA-secure scheme, we use the discrete-log-based, collision-resistant (and thus TCR) construct of [11] and show that it is also universal. However, some care needs to be taken about its choice of parameters, because the range of the hash must be "compatible" with the ABO TDF in our construction. Nevertheless, we demonstrate ways to achieve compatibility for two popular choices of groups where DDH is believed hard.

EXTENDING OUR GENERAL CONSTRUCTIONS. While our DDH-based instantiations fit neatly into a conceptual framework of "deterministic encryption with universal hash mode," they are not particularly efficient. Moreover, the other instantiations of lossy and ABO TDFs in [26] do *not* (at least immediately) give universal lossy modes. Our solution to this problem is to extend our general constructions in an efficient way such that the extra universality requirement on the underlying primitives is eliminated. These extensions derive from a novel application of a "crooked" version of the LHL due to Dodis and Smith [19], which tells us that if one applies an invertible, pairwise-independent hash function (e.g. the usual $H_{a,b}(x) = ax + b$ construct over a finite field) to a message before encrypting it under our general constructions, then "lossiness" of the underlying primitives (in addition to TCR for hash function H in the CCA case) alone suffices for security.

EFFICIENT PAILLIER-BASED SCHEMES. In particular, the above extensions allow us to instantiate our schemes using the "more-efficient" Paillier-based [25] lossy and ABO TDFs of [26]. However, these constructs are still far from optimal. Borrowing a technique of Damgård and Nielsen [14, 15], we devise new Paillier-based constructs of lossy and ABO TDFs having public-key size on the order of (instead of quadratic in) the message length and essentially no ciphertext expansion; moreover, they compare to standard Paillier encryption computationally. In order to encrypt messages with potentially-small min-entropy (relative to the length of a message), our constructs actually use a generalization of Paillier's scheme due to Damgård and Jurik [13]. Under this generalization, we also construct a hash function for H in the extended CCA-secure construction that is provably TCR based on the same assumption (decisional composite residuosity), and whose range is compatible with the ABO scheme. However, for practical efficiency one can instead use a TCR cryptographic hash function such as SHA256 or the constructs of [5, 29] for H. This is in fact another pleasing consequence of extending our general constructions, since before H was required to be both TCR and *universal*, which seems to preclude using a cryptographic hash function.

1.3 Concurrent Work

Concurrently and independently, Bellare, Fischlin, O'Neill and Ristenpart [3] define several multi-message, semantic-security-style definitions for deterministic encryption and prove them equivalent to PRIV definition of [1]. They also propose and prove equivalent an indistinguishability-based definition, but their proof techniques are different from ours. Namely, they consider an "intermediate" definitional variant that we do not. Also, they propose a new deterministic encryption scheme based on general assumptions, whereas our constructions are based on number-theoretic assumptions and are efficient. No constructions secure against chosen-ciphertext attacks are given in [3].

Our efficient Paillier-based instantiations of lossy and ABO TDFs were independently discovered by [27].

2 Preliminaries

Algorithms, probabilities and sources. Algorithms implicitly take as additional input the unary encoding 1^k of the security parameter k; they may be randomized and must run in poly-time in k unless indicated otherwise. Integer parameters are also implicitly polynomial functions of k. Adversaries are *non-uniform* and as such receive an auxiliary input of polynomial-size in k, which we also usually leave implicit. For a random variable Y, we write $y \stackrel{\$}{\leftarrow} Y$ to denote that y is sampled according to Y's distribution; furthermore, for an algorithm A, by $y \stackrel{\$}{\leftarrow} A(x)$ we mean that A is executed on input x and the output is assigned to y. (In the case that A gets no input we slightly abuse notation and write $y \stackrel{\$}{\leftarrow} A$ instead of $y \stackrel{\$}{\leftarrow} A()$.) We denote by $\Pr\big[A(x) = y : x \stackrel{\$}{\leftarrow} X\big]$ the probability that A outputs y on input x when x is sampled according to X. We say that an adversary A *has advantage* ϵ *in distinguishing* X *from* Y if $\Pr\big[A(x) = 1 : x \stackrel{\$}{\leftarrow} X\big]$ and $\Pr\big[A(y) = 1 : y \stackrel{\$}{\leftarrow} Y\big]$ differ by at most ϵ.

When more convenient, we use the following probability-theoretic notation instead. We write P_X for the distribution of random variable X and $P_X(x)$ for the probability that X puts on value x, i.e. $P_X(x) = \Pr[X = x]$. Similarly, we write $P_{X|\mathcal{E}}$ for the probability distribution of X conditioned on event $\mathcal{E}$, and P_{XY} for the joint distribution of random variables X, Y. The *statistical distance* between X and Y is given by $\Delta(X,Y) = \frac{1}{2}\sum_x |P_X(x) - P_Y(x)|$. It is well-known that if $\Delta(X,Y)$ is at most ϵ then any (even computationally unbounded) adversary A has advantage at most ϵ in distinguishing X from Y.

The *min-entropy* of a random variable X is $\mathrm{H}_\infty(X) = -\log(\max_x P_X(x))$. The *worst-case conditional* min-entropy of X given Y is defined as $\mathrm{H}_\infty(X|Y) = -\log(\max_{x,y} P_{X|Y=y}(x))$, and the *average conditional* min-entropy of X given Y as $\tilde{\mathrm{H}}_\infty(X|Y) = -\log(\sum_y P_Y(y) \max_x P_{X|Y=y}(x))$. A random variable X over $\{0,1\}^\ell$ is called a (t,ℓ)-*source* if $\mathrm{H}_\infty(X) \geq t$, and a list $\boldsymbol{X} = (X_1, \ldots, X_n)$ of random variables over $\{0,1\}^\ell$ is called a (t,ℓ)-*block-source* of length n if $\mathrm{H}_\infty(X_i|X_1 \ldots X_{i-1}) \geq t$ for all $i \in \{1, \ldots, n\}$.

A value $\nu \in \mathbb{R}$ depending on k is called *negligible* if its absolute value goes to 0 faster than any polynomial in k, i.e. $\forall\, c > 0\, \exists\, k_\circ \in \mathbb{N}\, \forall\, k \geq k_\circ : |\nu| < 1/k^c$.

Public-key encryption. An encryption scheme is a triple of algorithms $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, satisfying the usual syntax except that for convenience we also give the decryption algorithm $\mathcal{D}$ the public key. For simplicity, we only consider a message-space of $\{0,1\}^\ell$, and we say that $\mathcal{AE}$ is an ℓ-*bit encryption scheme* if for all messages in the message space $\{0,1\}^\ell$

$$\Pr\big[\, \mathcal{D}(sk, \mathcal{E}(pk, m)) \neq m :\ (pk, sk) \xleftarrow{\$} \mathcal{K} \,\big]$$

is negligible. We say that $\mathcal{AE}$ is *deterministic* if $\mathcal{E}$ is deterministic. Note that we require the message-space to depend only on the security parameter and not on the specific public key; as in [1] this is somewhat crucial to our security definitions.

Hashing. An ℓ-*bit hash function* $\mathcal{H} = (\mathcal{K}, H)$ with domain $\{0,1\}^\ell$ consists of a key-generation algorithm and a hash algorithm.[3] Again, we omit the well-known syntax and restrict to domain $\{0,1\}^\ell$ for simplicity. We say $\mathcal{H}$ has a 2^r-*bounded hash range* if its range $R = \{H(K,x) \mid K \in \mathcal{K}, x \in D\}$ is bounded by $|R| \leq 2^r$ in size. We say that $\mathcal{H}$ with range R is *universal* if for all $x_1 \neq x_2 \in \{0,1\}^\ell$

$$\Pr\big[\, H(K,x_1) = H(K,x_2) :\ K \xleftarrow{\$} \mathcal{K} \,\big] \ \leq\ \frac{1}{|R|}\ ,$$

and we say it is *pairwise-independent* if for all $x_1 \neq x_2 \in \{0,1\}^\ell$ and all $y_1, y_2 \in R$

$$\Pr\big[\, H(K,x_1) = y_1 \ \wedge\ H(K,x_2) = y_2 :\ K \xleftarrow{\$} \mathcal{K} \,\big] \ \leq\ \frac{1}{|R|^2}\ .$$

We say $\mathcal{H}$ is *collision-resistant* (CR) if for every poly-time A the *CR-advantage*

$$\mathbf{Adv}^{\mathrm{cr}}_{\mathcal{H}}(A) \ =\ \Pr\big[\, H(K,x_1) = H(K,x_2) :\ K \xleftarrow{\$} \mathcal{K}\, ;\ (x_1,x_2) \xleftarrow{\$} A(K) \,\big]$$

is negligible. Similarly, we say $\mathcal{H}$ is *target-collision resistant* (TCR) if for every poly-time A the *TCR-advantage*

$$\mathbf{Adv}^{\mathrm{tcr}}_{\mathcal{H}}(A) = \Pr\big[\, H(K,x_1) = H(K,x_2) :\ (x_1, \mathrm{st}) \xleftarrow{\$} A\, ;\ K \xleftarrow{\$} \mathcal{K}\, ;\ x_2 \xleftarrow{\$} A(K, \mathrm{st}) \,\big]$$

[3] Note that we are not only interested in "compressing" hash functions, e.g. images and pre-images might have the same bit-length.

is negligible. As discussed in [5] TCR has some potential benefits over CR, such as being easier to achieve and allowing for shorter output lengths.

3 Security Definitions

The PRIV notion of security for deterministic encryption introduced in [1] asks that it be hard to guess any partial information[4] of a list of messages given their encryptions, as long as the list has component-wise high (super-logarithmic) min-entropy. We introduce a slight weakening of this notion where each message must have high min-entropy *conditioned on values of the other messages.* This notion seems to nevertheless suffice in some practical applications, for example in the encryption of high-entropy data containing phone or social security numbers that can share prefixes but are otherwise uncorrelated. We then consider two other security definitions in order of increasing simplicity and ease-of-use; in the next section we prove that they are all equivalent.

PRIV FOR BLOCK-SOURCES. The following is a semantic-security-style definition that considers the encryption of multiple messages under the same public-key. For an ℓ-bit encryption scheme $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ and list $\boldsymbol{m} = (m_1, \ldots, m_n)$ of messages, we write $\boldsymbol{\mathcal{E}}(pk, \boldsymbol{m})$ below as shorthand for $(\mathcal{E}(pk, m_1), \ldots, \mathcal{E}(pk, m_n))$.

Definition 1. *An ℓ-bit encryption scheme $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is* PRIV-secure for (t, ℓ)-block-sources *if for any (t, ℓ)-block-source $\boldsymbol{M} = (M_1, \ldots, M_n)$ of polynomial length n, any function $f : \{0,1\}^{n\ell} \to \{0,1\}^*$ and all poly-time adversaries A, the PRIV-advantage*

$$\mathbf{Adv}^{\mathrm{priv}}_{\mathcal{AE}}(A, f, \boldsymbol{M}) \;=\; \mathbf{Real}_{\mathcal{AE}}(A, f, \boldsymbol{M}) - \mathbf{Ideal}_{\mathcal{AE}}(A, f, \boldsymbol{M})$$

is negligible, where

$$\mathbf{Real}_{\mathcal{AE}}(A, f, \boldsymbol{M}) = \Pr\big[A(pk, \boldsymbol{\mathcal{E}}(pk, \boldsymbol{m})) = f(\boldsymbol{m}) : (pk, sk) \xleftarrow{\$} \mathcal{K}\,;\ \boldsymbol{m} \xleftarrow{\$} \boldsymbol{M}\big] \text{ and}$$
$$\mathbf{Ideal}_{\mathcal{AE}}(A, f, \boldsymbol{M}) = \Pr\big[A(pk, \boldsymbol{\mathcal{E}}(pk, \boldsymbol{m'})) = f(\boldsymbol{m}) : (pk, sk) \xleftarrow{\$} \mathcal{K};\ \boldsymbol{m}, \boldsymbol{m'} \xleftarrow{\$} \boldsymbol{M}\big]$$

A SINGLE-MESSAGE DEFINITION. Consider Definition 1 with the restriction that only (t, ℓ)-block-sources of length $n = 1$ are allowed; that is, a (t, ℓ)-source M replaces block-source $\boldsymbol{M}$ in the definition. Call the resulting notion *PRIV1-security for (t, ℓ)-sources*, where we define $\mathbf{Real}_{\mathcal{AE}}(A, f, M)$ and $\mathbf{Ideal}_{\mathcal{AE}}(A, f, M)$ as well as the PRIV1-advantage $\mathbf{Adv}^{\mathrm{priv1}}_{\mathcal{AE}}(A, f, M)$ accordingly.

We note that (an alternative formulation of) PRIV1 was already considered in [1], and it was shown to be strictly weaker than their multi-message notion PRIV. We will show that in the setting of *block*-sources the single- and multi-message definitions are equivalent.

[4] To make the definition achievable, the partial information must not depend on the public key. This is reasonable since real data does not depend on any public key.

AN INDISTINGUISHABILITY-BASED FORMULATION. We also consider the following indistinguishability-based formulation of PRIV1 inspired by [20], which is handier to work with. It asks that it be hard to distinguish the encryptions of two plaintexts, each drawn from a different (public-key-independent) high-entropy distribution on the message-space.

Definition 2. *An ℓ-bit encryption scheme $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is* PRIV1-IND *-secure for (t, ℓ)-sources if for any (t, ℓ)-sources M_0 and M_1 and all poly-time adversaries A, the* PRIV1-IND-advantage

$$\mathbf{Adv}^{\text{priv1-ind}}_{\mathcal{AE}}(A, M_0, M_1) \;=\; \mathbf{Guess}_{\mathcal{AE}}(A, M_0) - \mathbf{Guess}_{\mathcal{AE}}(A, M_1)$$

is negligible, where for $b \in \{0, 1\}$

$$\mathbf{Guess}_{\mathcal{AE}}(A, M_b) \;=\; \Pr\big[A(pk, \mathcal{E}(pk, m_b)) = 1 : \, (pk, sk) \xleftarrow{\$} \mathcal{K} \,;\, m_b \xleftarrow{\$} M_b\big]\,.$$

We note that concurrently and independently, [3] gives an indistinguishability-based formulation of the multi-message PRIV definition from [1] (that does not restrict to block-sources).

EXTENSION TO CHOSEN-CIPHERTEXT ATTACKS (CCA). For simplicity, the presented definitions only consider the case of chosen-plaintext attacks (CPA).[5] To extend the definitions to the *chosen-ciphertext-attack* (CCA) setting, we can additionally provide the adversary A in each definition with access to decryption oracle $\mathcal{D}(pk, sk, \cdot)$, which it may query on any ciphertext not appearing in its input. We denote the resulting notions with "-CCA" (e.g. PRIV-CCA for block-sources). Our equivalence results in the following also hold in the CCA setting.

Remark 1. The PRIV definition (and similarly the PRIV1 definition) in [1] requires the pair $(\boldsymbol{m}, s)$ of message-list $\boldsymbol{m}$ and partial-information s on $\boldsymbol{m}$ to be *poly-time samplable*. We do not have such restrictions in our definitions. On the other hand, we ask s to be a *deterministic* function $s = f(\boldsymbol{m})$ of $\boldsymbol{m}$; this latter restriction, however, is without loss of generality, as we argue in Remark 1 below (as long as we allow f to be unbounded). Thus, our definitions remain at least as strong as their corresponding formulations in the style of [1]. The reason for omitting samplability restrictions is for generality and to simplify our results and proofs, and because they are actually not required for the security of our constructions. Furthermore, this strengthening of the definitions is not crucial for our equivalence results; see Remark 4.

Remark 2. PRIV1 (similarly PRIV for block-sources) remains equivalent if we allow f to be *randomized*; i.e., on input m the function f is evaluated as $f(m; r)$ for r chosen independently according to some fixed probability distribution (typically uniform) on a finite domain. This equivalence holds for both the "private

[5] Actually, the plaintexts themselves in the definitions are not chosen by the adversary. This is a minor semantic point that we ignore.

seed" model, where adversary A does not learn r, and the "public coin" model, where r is given to A (or in a combination of the two). Indeed, if for some adversary, randomized function and block-source, the advantage of A is in absolute value lower-bounded by ε *on average* over the random choice of r, then the same lower-bound holds for some specific choice of r. (The other direction is trivial.)

Note that the "private seed" model covers the case, similar to [1], where a message-and-partial-info pair (m, s) is chosen according to an *arbitrary joint probability distribution* P_{MS} (with $\mathrm{H}_\infty(M) \geq t$ and a finite domain for s), as we can always understand the message m as instead sampled according to its distribution P_M and then the partial-information s computed with conditional distribution $P_{S|M=m}$ by means of a randomized function (which can always be done since we do not require f to be efficient[6]). Thus, if in the "private seed" model we restrict the message-and-partial-info pair to be poly-time samplable, then our PRIV1 definition is equivalent to that from [1].

Remark 3. It also suffices in the above definitions to consider *predicates* f, i.e., binary functions to $\{0, 1\}$. This actually follows from Lemma 3 of [16] (and verifying that their proof also works in our poly-time-adversary setting). The idea is to consider the Goldreich-Levin (i.e. inner-product) predicate of the partial information with a random string, and use Remark 2. The resulting adversary loses a factor 2 in its advantage and its running-time increases by $O(n\ell)$. (The technique also works for definitions in the style of [1]; i.e., it suffices to consider partial information of length 1 there.)

4 Equivalence of the Definitions

We show that all three definitions, namely PRIV for block-sources, PRIV1 and PRIV1-IND, are equivalent. Our strategy is as follows. We take PRIV1 as our starting point, and we first show that it is equivalent to PRIV1-IND. Later we show that it is also equivalent to PRIV for block-sources.

Theorem 1. *Let $\mathcal{AE}$ be an ℓ-bit encryption scheme. Then for any (t, ℓ)-sources M_0, M_1 and any adversary A, there exists a (t, ℓ)-source M, an adversary B and a function f such that*

$$\mathbf{Adv}^{\text{priv1-ind}}_{\mathcal{AE}}(A, M_0, M_1) \;\leq\; 2 \cdot \mathbf{Adv}^{\text{priv1}}_{\mathcal{AE}}(B, f, M)\;,$$

and the running-time of B is that of A. And, for any $(t+1, \ell)$-source M, any function f and any adversary A, there exists an adversary B and (t, ℓ)-sources M_0, M_1 such that

$$\mathbf{Adv}^{\text{priv1}}_{\mathcal{AE}}(A, f, M) \;\leq\; 2 \cdot \mathbf{Adv}^{\text{priv1-ind}}_{\mathcal{AE}}(B, M_0, M_1)\;,$$

and the running-time of B is that of A plus $O(\ell)$. □

[6] E.g., r could consist of a list of suitable choices for s, one choice for each possible m, and f would select and output the right entry.

The proof borrows and combines ideas from [20] and [16, 17], used for showing the equivalence between entropical security for information-theoretic symmetric (quantum) encryption schemes and an indistinguishability-based notion.[7]

The proof of the first claim relies on Remark 2 and is straightforward, since distinguishing M_0, M_1 given their encryptions is equivalent to guessing b from the encryption of M_b where b is a random bit. For the second claim, note that if $f(M)$ is easy-to-guess given the encryption of M, then M conditioned on $f(M) = 0$ and M conditioned on $f(M) = 1$ are easy to distinguish. However, one of these distributions may have much smaller min-entropy than M (if f is unbalanced). To avoid (almost all of) this entropy loss we can "mix" them appropriately with M. Moreover, the resulting distributions become poly-time samplable if the pair $(M, f(M))$ is (see Remark 4).

Proof. We start with the first claim. Let M_0, M_1 and A be as given. Let M to be the balanced "mixture" of M_0 and M_1, and f be the corresponding "indicator function;" i.e., M is sampled by choosing a random bit b and then outputting m sampled according to M_b, and the partial information $f(m)$ is defined as b. Such a joint probability distribution on m and b is allowed by Remark 2. Let B be the PRIV1-adversary that on inputs pk, c runs A on the same inputs and outputs the result. Then $\mathrm{H}_\infty(M) \geq t$ and we have

$$\begin{aligned}
\mathbf{Adv}^{\text{priv1}}_{\mathcal{AE}}(B, f, M) &= \mathbf{Real}_{\mathcal{AE}}(B, f, M) - \mathbf{Ideal}_{\mathcal{AE}}(B, f, M) \\
&= \Big(\frac{1}{2}(1 - \mathbf{Guess}_{\mathcal{AE}}(A, M_0)) + \frac{1}{2}\mathbf{Guess}_{\mathcal{AE}}(A, M_1)\Big) - \frac{1}{2} \\
&= \frac{1}{2}\big(\mathbf{Guess}_{\mathcal{AE}}(A, M_1) - \mathbf{Guess}_{\mathcal{AE}}(A, M_0)\big) \\
&= \frac{1}{2}\mathbf{Adv}^{\text{priv1-ind}}_{\mathcal{AE}}(A, M_0, M_1)\,;
\end{aligned}$$

this proves the first claim.

For the second claim, let A, f, M be as given. We first note that by Remark 3, we may assume that $f : \{0,1\}^\ell \to \{0,1\}$, at the cost of losing at most a factor 2 in A's advantage and increasing its running-time by $O(\ell)$. Consider the independent random variables M_0 and M_1, with respective distributions

$$P_{M_0} = r_0 P_{M|f(M)=0} + r_1 P_M \qquad \text{and} \qquad P_{M_1} = r_1 P_{M|f(M)=1} + r_0 P_M \;,$$

where $r_0 = P_{f(M)}(0)$ and $r_1 = P_{f(M)}(1)$. Then for any $m \in \{0,1\}^\ell$

$$\begin{aligned}
P_{M_0}(m) &= r_0 P_{M|f(M)=0}(m) + r_1 P_M(m) = P_{Mf(M)}(m, 0) + r_1 P_M(m) \\
&\leq 2^{-t-1} + r_1 2^{-t-1} \leq 2^{-t} \;,
\end{aligned}$$

[7] Note that the definition of entropic security may come in different flavors, named *ordinary* and *strong* in [16]. The (ordinary) notion used in [20] makes their proof much more cumbersome since Remark 3 does not apply (directly). Our definition of PRIV corresponds to the *strong* flavor.

and similarly $P_{M_1}(m) \leq 2^{-t}$, so that $\mathrm{H}_\infty(M_0), \mathrm{H}_\infty(M_1) \geq t$ as required. Let B be the PRIV1-IND adversary that runs the same code as A. It remains to argue that B can distinguish M_0 and M_1. In order to simplify notation, we let Y, Y_0 and Y_1 be the random variables defined by $Y = A(PK, \mathcal{E}(PK, M))$, $Y_0 = A(PK, \mathcal{E}(PK, M_0))$ and $Y_1 = A(PK, \mathcal{E}(PK, M_1))$, where PK describes a public key generated by $\mathcal{K}$.[8] We have

$$\begin{aligned}
&\mathbf{Adv}^{\text{priv1-ind}}_{\mathcal{AE}}(B, M_0, M_1) \;=\; \mathbf{Guess}_{\mathcal{AE}}(B, M_1) - \mathbf{Guess}_{\mathcal{AE}}(B, M_0) \\
&\quad = P_{Y_1}(1) - P_{Y_0}(1) \;=\; P_{Y_1}(1) - (1 - P_{Y_0}(0)) \;=\; P_{Y_1}(1) + P_{Y_0}(0) - 1 \;, \qquad (1)
\end{aligned}$$

where the second equality is by construction. Note that $P_{Y_0} = r_0 P_{Y|f(M)=0} + r_1 P_Y$ and similarly for P_{Y_1}. It follows that

$$\begin{aligned}
&P_{Y_0}(0) + P_{Y_1}(1) \\
&\;=\; \big(r_0 P_{Y|f(M)=0}(0) + r_1 P_Y(0)\big) + \big(r_1 P_{Y|f(M)=1}(1) + r_0 P_Y(1)\big) \\
&\;=\; \big(r_0 P_{Y|f(M)=0}(0) + r_1 P_{Y|f(M)=1}(1)\big) + \big(r_0 P_Y(1) + r_1 P_Y(0)\big) \\
&\;=\; \big(r_0 P_{Y|f(M)=0}(0) + r_1 P_{Y|f(M)=1}(1)\big) + 1 - \big(r_0 P_Y(0) + r_1 P_Y(1)\big) \\
&\;=\; \big(P_{Y f(M)}(0,0) + P_{Y f(M)}(1,1)\big) + 1 - \big(P_{f(M)}(0) P_Y(0) + P_{f(M)}(1) P_Y(1)\big) \\
&\;=\; \Pr[Y = f(M)] - \Pr[Y = f(M')] + 1 \\
&\;=\; \mathbf{Real}_{\mathcal{AE}}(A, f, M) - \mathbf{Ideal}_{\mathcal{AE}}(A, f, M) + 1 \;=\; \mathbf{Adv}^{\text{priv1}}_{\mathcal{AE}}(A, f, M) + 1 \;,
\end{aligned}$$

where M' is an independent identically-distributed copy of M. Note that we use $r_0 + r_1 = 1$ and $P_Y(0) + P_Y(1) = 1$ in the third equality and in the second-to-last we use that we can switch the roles of m and m' in the definition of $\mathbf{Ideal}_{\mathcal{AE}}(A, f, M)$. Substituting into equation (1), we obtain

$$\mathbf{Adv}^{\text{priv1-ind}}_{\mathcal{AE}}(B, M_0, M_1) \;=\; \mathbf{Adv}^{\text{priv1}}_{\mathcal{AE}}(A, f, M) \;.$$

Taking into account the factor-2 loss, this proves the second claim. □

Next, we show that PRIV1 for (t, ℓ)-sources implies PRIV for (t, ℓ)-block-sources; the reverse implication holds trivially.

Theorem 2. *Let $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an ℓ-bit encryption scheme. For any (t, ℓ)-block-source $\boldsymbol{M}$ of length n, any function $f : \{0,1\}^{n\ell} \to \{0,1\}^*$ and any adversary A, there exists a (t, ℓ)-source M, a function g and an adversary B such that*

$$\mathbf{Adv}^{\text{priv1}}_{\mathcal{AE}}(A, \boldsymbol{M}, f) \;\leq\; 10n \cdot \mathbf{Adv}^{\text{priv}}_{\mathcal{AE}}(B, M, g) \;.$$

Furthermore, the running-time of B is at most that of A plus $O(n\ell)$. □

Interestingly, the proof is not a straightforward hybrid argument, but makes intensive use of Theorem 1. The idea is to consider the probability of the adversary

[8] It makes no difference for the upcoming argument whether we consider the same or a fresh public key for Y, Y_0 and Y_1.

A in guessing $f(\boldsymbol{M})$ when given the encryption of a list of *independent* and *uniformly* distributed messages and compare this both to $\mathbf{Ideal}_{\mathcal{AE}}(A, f, \boldsymbol{M})$ and to $\mathbf{Real}_{\mathcal{AE}}(A, f, \boldsymbol{M})$, making use of hybrid arguments and the PRIV1-IND-security of $\mathcal{AE}$ (which follows from its assumed PRIV1-security).

Proof. Let $A, \boldsymbol{M}, f$ be as given. By Remark 3, we may assume that f is *binary*, at the cost of losing a factor 2 in A's advantage and increasing its running-time by $O(n\ell)$. Furthermore, we may assume the PRIV1-advantage to be non-negative (otherwise we flip A's output bit). To simplify notation, we write $\mathbf{Adv}(A)$ below as shorthand for $\mathbf{Adv}^{\text{priv1}}_{\mathcal{AE}}(A, \boldsymbol{M}, f)$. Consider the probability

$$\mathbf{u}_{\mathcal{AE}}(A, \boldsymbol{M}, f) = \Pr\big[A(pk, \boldsymbol{\mathcal{E}}(pk, \boldsymbol{u})) = f(\boldsymbol{m}) : (pk, sk) \leftarrow \mathcal{K}\ ;\ \boldsymbol{m} \leftarrow \boldsymbol{M}\ ;\ \boldsymbol{u} \leftarrow \boldsymbol{U}\big]$$

with $\boldsymbol{U} = (U_1, \ldots, U_n)$ being n independent copies of the uniform distribution on $\{0,1\}^\ell$. Note that we can re-write $\mathbf{Adv}(A)$ as

$$\big(\mathbf{Real}_{\mathcal{AE}}(A, f, \boldsymbol{M}) - \mathbf{u}_{\mathcal{AE}}(A, f, \boldsymbol{M})\big) + \big(\mathbf{u}_{\mathcal{AE}}(A, f, \boldsymbol{M}) - \mathbf{Ideal}_{\mathcal{AE}}(A, f, \boldsymbol{M})\big).$$

Intuitively, this implies that if $\mathbf{Adv}(A)$ is "large" then one of the above two summands must be as well. We show that in either case we can construct a (t, ℓ)-source M, a function g and an adversary B as claimed. We start with the latter case. Specifically, suppose that

$$\mathbf{u}_{\mathcal{AE}}(A, f, \boldsymbol{M}) - \mathbf{Ideal}_{\mathcal{AE}}(A, f, \boldsymbol{M}) \ \geq\ \frac{2}{5}\mathbf{Adv}(A)\ .$$

We construct a PRIV1-IND adversary B with running-time that of A plus $O(n\ell)$ and two (t, ℓ)-sources with resulting PRIV1-IND advantage lower bounded by $2\mathbf{Adv}(A)/5n$; Theorem 1 then implies the claim (taking into account the factor-2 loss by our initial assumption that f is binary). We use a hybrid argument. For $i \in \{0, \ldots, n\}$ consider the probability

$$\mathbf{h}^{1,i}_{\mathcal{AE}}(A, f, \boldsymbol{M}) = \Pr\left[\begin{array}{r} A(pk, \boldsymbol{\mathcal{E}}(pk, (m'_1, \ldots, m'_i, u_{i+1}, \ldots, u_n))) = f(\boldsymbol{m}) : \\ (pk, sk) \xleftarrow{\$} \mathcal{K}\ ;\ \boldsymbol{m}, \boldsymbol{m'} \xleftarrow{\$} \boldsymbol{M}, \boldsymbol{u} \xleftarrow{\$} \boldsymbol{U} \end{array}\right].$$

It follows that there exists a j such that $\mathbf{h}^{1,j}_{\mathcal{AE}}(A, f, \boldsymbol{M}) - \mathbf{h}^{1,j+1}_{\mathcal{AE}}(A, f, \boldsymbol{M})$ is at least $2\mathbf{Adv}(A)/5n$. Furthermore, this lower-bound holds for some specific choices $\dot{m}'_1, \ldots, \dot{m}'_j$ of $m'_1, \ldots m'_j$ and some specific choice $\dot{\boldsymbol{m}}$ of $\boldsymbol{m}$. We assume for simplicity that $f(\dot{\boldsymbol{m}}) = 1$; if it is 0 the argument is similar. This implies that there exists an adversary B, which on inputs pk, c samples $\boldsymbol{u} \xleftarrow{\$} \boldsymbol{U}$ and returns

$$A(pk, \mathcal{E}(pk, \dot{m}'_1), \ldots, \mathcal{E}(pk, \dot{m}'_j), c, \mathcal{E}(pk, u_{j+2}), \ldots, \mathcal{E}(pk, u_n))\ ,$$

and two (t, ℓ)-sources, namely M_{j+1} conditioned on $M_1 = \dot{m}'_1, \ldots, M_j = \dot{m}'_j$ and U_{j+1}, such that the resulting PRIV1-IND advantage is lower bounded by $2\mathbf{Adv}(A)/5n$, as required.

We move to the other case, where we have

$$\mathbf{Real}_{\mathcal{AE}}(A, f, \boldsymbol{M}) - \mathbf{u}_{\mathcal{AE}}(A, f, \boldsymbol{M}) \ \geq\ \frac{3}{5}\mathbf{Adv}(A)\ .$$

We use another hybrid argument. Specifically, for $i \in \{0,\ldots,n\}$ consider the probability

$$\mathbf{h}^{2,i}_{\mathcal{AE}}(A,f,\boldsymbol{M}) = \Pr\left[\begin{array}{r} A(pk,\boldsymbol{\mathcal{E}}(pk,(m_1,\ldots,m_i,u_{i+1},\ldots,u_n))) = f(\boldsymbol{m}) : \\ (pk,sk) \xleftarrow{\$} \mathcal{K}\,;\ \boldsymbol{m} \xleftarrow{\$} \boldsymbol{M}, \boldsymbol{u} \xleftarrow{\$} \boldsymbol{U} \end{array}\right].$$

Again it follows that there exists a j such that $\mathbf{h}^{2,j+1}_{\mathcal{AE}}(A,f,\boldsymbol{M}) - \mathbf{h}^{2,j}_{\mathcal{AE}}(A,f,\boldsymbol{M})$ is at least $3\mathbf{Adv}(A)/5n$, and that this lower-bound holds for some specific choices $\dot{m}_1,\ldots,\dot{m}_j$ of $m_1,\ldots,m_j$. Let us denote the corresponding probabilities with these choices by $\dot{\mathbf{h}}^{2,j+1}_{\mathcal{AE}}(A,f,\boldsymbol{M})$ and $\dot{\mathbf{h}}^{2,j}_{\mathcal{AE}}(A,f,\boldsymbol{M})$. Consider now the (t,ℓ)-source M with distribution $P_M = P_{M_{j+1}|M_1=\dot{m}_1,\ldots,M_j=\dot{m}_j}$. By assumption we have $\mathrm{H}_\infty(M) \geq t$. Also, consider the "randomized" function g (in the "private seed" model) defined as

$$g(m; m_{j+2},\ldots,m_n) \;=\; f(\dot{m}_1,\ldots,\dot{m}_j, m, m_{j+2},\ldots,m_n)\;,$$

with $m_{j+2},\ldots,m_n$ chosen according to the distribution of $M_{j+2},\ldots,M_n$, conditioned on $M_1 = \dot{m}_1,\ldots,M_j = \dot{m}_j$ and $M_{j+1} = m$. By Remark 2, it indeed suffices to consider such a function. Let B be the PRIV1 adversary that on input pk, c, samples $\boldsymbol{u} \xleftarrow{\$} \boldsymbol{U}$ and outputs

$$A\big(pk, \mathcal{E}(pk,\dot{m}_1),\ldots,\mathcal{E}(pk,\dot{m}_j), c, \mathcal{E}(pk,u_{j+2}),\ldots,\mathcal{E}(pk,u_k)\big).$$

Now by construction, $\mathbf{Real}_{\mathcal{AE}}(B,g,M)$ coincides with $\dot{\mathbf{h}}^{2,j+1}_{\mathcal{AE}}(A,f,\boldsymbol{M})$ and thus $\mathbf{Real}_{\mathcal{AE}}(B,g,M) - \dot{\mathbf{h}}^{2,j}_{\mathcal{AE}}(A,f,\boldsymbol{M}) \geq 3\mathbf{Adv}(A)/5n$. We consider two cases. If $\dot{\mathbf{h}}^{2,j}_{\mathcal{AE}}(A,f,\boldsymbol{M}) - \mathbf{Ideal}_{\mathcal{AE}}(B,g,M) \geq \mathbf{Adv}(A)/5n$ then the claim follows. Otherwise, $\dot{\mathbf{h}}^{2,j}_{\mathcal{AE}}(A,f,\boldsymbol{M}) - \mathbf{Ideal}_{\mathcal{AE}}(B,g,M)$ is at least $2\mathbf{Adv}(A)/5n$. Then this lower-bound also holds for some particular choices $\dot{m}_{j+1},\ldots,\dot{m}_n$ of $m_{j+1},m_{j+2},\ldots,m_n$ in the definition of $\dot{\mathbf{h}}^{2,j}_{\mathcal{AE}}(A,f,\boldsymbol{M})$ and the same choices of $m,m_{j+2},\ldots,m_n$ in the definition of $\mathbf{Ideal}_{\mathcal{AE}}(B,g,M)$. Let us denote the corresponding probabilities with these choices by $\ddot{\mathbf{h}}^{2,j}_{\mathcal{AE}}(A,f,\boldsymbol{M})$ and $\mathbf{Id\dot{e}al}_{\mathcal{AE}}(B,g,M)$. Furthermore, let us assume for simplicity that $f(\dot{m}_1,\ldots,\dot{m}_n) = 1$. Then re-using B as a PRIV1-IND adversary, by construction $\mathbf{Guess}_{\mathcal{AE}}(B,U_{j+1}) = \ddot{\mathbf{h}}^{2,j}_{\mathcal{AE}}(A,f,\boldsymbol{M})$ and $\mathbf{Guess}_{\mathcal{AE}}(B,M) = \mathbf{Id\dot{e}al}_{\mathcal{AE}}(B,g,M)$, so the claim follows by Theorem 1 (though now with different choices of B,g,M in the statement). $\square$

Remark 4. Our proof of Theorem 1 also works if as in [1] we require message-and-partial-info pairs (M,S) in the PRIV1 definition, and message-sources M_0 and M_1 in the PRIV1-IND definition to be *poly-time samplable* (allowing S to depend probabilistically on M). Indeed, in the proof of the first claim, note that if M_0 and M_1 are poly-time samplable then so is the pair (M_B,B) where B is a random bit. In the second, note that if the message-and-partial-info pair (M,S), where S is a bit, is poly-time samplable then the following is a poly-time sampler for M_0 (the sampler for M_1 is symmetric): Sample (m,s) and output m if $s=0$; else, sample (m',s') and output m'. (Specifically the running-time of the sampler is at most twice that of the original one in this case.) As such, essentially the same proof can be used to obtain equivalence between the multi-message PRIV and IND definitions shown in [3] as well.

Similarly, our proof of Theorem 2 also works when restricting to such poly-time samplable message-and-partial-info pairs, where though in the PRIV definition for block-sources we need that $(\boldsymbol{M}, S)$ can be sampled by a poly-time algorithm *conditioned on any fixed choice for* $M_1, \ldots, M_j$ and for any j. Indeed, in the reduction we fix a particular choice $\dot{m}_1, \ldots, \dot{m}_j$ for $M_1, \ldots, M_j$ (for some j) and construct a PRIV1 adversary based upon the message-and-partial-info pair (M_{j+1}, S) conditioned on $(M_1, \ldots, M_j) = (\dot{m}_1, \ldots, \dot{m}_j)$. This is poly-time samplable under the above samplability condition on $(\boldsymbol{M}, S)$.

5 General CPA- and CCA-Secure Constructions

We propose general constructions of deterministic encryption that are CPA- and CCA-secure under our security notions. The constructions derive from an interesting connection between deterministic encryption and "lossy" trapdoor functions introduced by Peikert and Waters [26]. These are trapdoor functions with a (un-invertible) "lossy" mode in which the function loses information about its input, and for which the outputs of the "normal" and "lossy" modes are (computationally) indistinguishable. Viewing trapdoor functions as deterministic encryption schemes in our context, we develop a similar framework of *deterministic encryption with hidden hash mode.*

5.1 CPA-Secure Construction

For our CPA-secure construction, we introduce the following notion.

Deterministic Encryption with Hidden Hash Mode. We say that $\mathcal{AE} = (\mathcal{K}, \tilde{\mathcal{K}}, \mathcal{E}, \mathcal{D})$ is a deterministic ℓ-bit encryption scheme with *hidden hash mode* (HHM), or simply HHM deterministic encryption scheme, with a 2^r*-bounded hash range* if $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ is an ℓ-bit deterministic encryption scheme, and the following conditions are satisfied:

- (*Algorithm* $\tilde{\mathcal{K}}$ *induces a hash.*) There is an induced hash function $\mathcal{H}_{\mathcal{E}} = (\tilde{\mathcal{K}}, H_{\mathcal{E}})$ with domain $\{0,1\}^\ell$ and a 2^r-bounded hash range, where algorithm $\tilde{\mathcal{K}}$ outputs $\tilde{pk}$, and $H_{\mathcal{E}}$ on inputs $\tilde{pk}, m$ returns $\mathcal{E}(\tilde{pk}, m)$. (Typically $r \ll \ell$.)
- (*Hard to tell* $\tilde{pk}$ *from* pk.) Any poly-time adversary A has negligible advantage, denoted $\mathbf{Adv}^{\mathrm{hhm}}_{\mathcal{AE}}(A)$, in distinguishing the first outputs of $\tilde{\mathcal{K}}$ and $\mathcal{K}$.

The "alternate" key-generation algorithm $\tilde{\mathcal{K}}$ is used only for security proofs; we assume it produces only a public key and no secret key. In the case that the induced encryption scheme $\mathcal{H}_{\mathcal{E}}$ in the first property is universal, we say that scheme $\mathcal{AE}$ has a *hidden universal-hash mode* (HUHM).

HUHM implies CPA-security. We show that a deterministic encryption scheme with hidden universal-hash mode is in fact PRIV-secure for block-sources. In other words, if the lossy mode of a lossy trapdoor function is universal, then it is a CPA-secure deterministic encryption scheme in our sense.

Theorem 3. *Let* $\mathcal{AE} = (\mathcal{K}, \tilde{\mathcal{K}}, \mathcal{E}, \mathcal{D})$ *be an* ℓ*-bit deterministic encryption scheme with a HUHM and a* 2^r*-bounded hash range. Then for any adversary* A*, any* (t, ℓ)*-sources* M_0, M_1 *and any* $\epsilon > 0$ *such that* $t \geq r + 2\log(1/\epsilon)$*, there exists an adversary* B *such that*

$$\mathbf{Adv}_{\mathcal{AE}}^{\text{priv1-ind}}(A, M_0, M_1) \;\leq\; 2 \cdot \left(\mathbf{Adv}_{\mathcal{AE}}^{\text{hhm}}(B) + \epsilon\right) .$$

Furthermore, the running-time of B *is that of* A. □

The idea of the proof is simple: in the experiments for the PRIV1-IND adversary A, the alternate key generation algorithm $\tilde{\mathcal{K}}$ of $\mathcal{AE}$ may be used instead of $\mathcal{K}$ without A being able to tell the difference; then, the Leftover Hash Lemma (LHL) [23, 6] implies that "encryptions" are essentially uniform, so it is impossible for A to guess from which source the encrypted message originated. (Note that it is not crucial here that the output distribution be *uniform*, but merely independent of the input distribution.)

Proof. For $b \in \{0, 1\}$, by definition of $\mathbf{Adv}_{\mathcal{AE}}^{\text{hhm}}$, the probability

$$\mathbf{Guess}_{\mathcal{AE}}(A, M_b) \;=\; \Pr\big[A(pk, \mathcal{E}(pk, m_b)) = 1 :\; (pk, sk) \xleftarrow{\$} \mathcal{K} \,;\; m_b \xleftarrow{\$} M_b\big]$$

differs from the probability

$$\Pr\big[A(\tilde{pk}, \mathcal{E}(\tilde{pk}, m_b)) = 1 :\; \tilde{pk} \xleftarrow{\$} \tilde{\mathcal{K}} \,;\; m_b \xleftarrow{\$} M_b\big]$$

by at most $\sum_m P_{M_b}(m)\, \mathbf{Adv}_{\mathcal{AE}}^{\text{hhm}}(B_m)$, where B_m on any input pk simply runs and outputs $A(pk, \mathcal{E}(pk, m))$. By the universal property of the hash mode and applying the LHL, it follows that the above probability is within ϵ of

$$\Pr\big[A(\tilde{pk}, c) = 1 :\; \tilde{pk} \xleftarrow{\$} \tilde{\mathcal{K}} \,;\; c \xleftarrow{\$} R\big]$$

where R denotes the range of the induced hash function $\mathcal{H}_{\mathcal{E}}$. But now, this probability does not depend on b anymore, and thus

$$\mathbf{Adv}_{\mathcal{AE}}^{\text{priv1-ind}}(A, M_0, M_1) \;\leq\; \sum_m \big(P_{M_0}(m) + P_{M_1}(m)\big) \mathbf{Adv}_{\mathcal{AE}}^{\text{hhm}}(B_m) + 2\epsilon$$

from which the claim follows by a suitable choice of m. □

5.2 CCA-Secure Construction

In order to extend the connection between lossy TDFs and deterministic encryption to the CCA setting, we first generalize our notion of deterministic encryption with HHM in a similar way to the all-but-one (ABO) TDF primitive defined in [26].

All-but-one deterministic encryption. An *all-but-one* (ABO) deterministic encryption scheme $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ with a 2^r-bounded hash range is such that each of $\mathcal{K}, \mathcal{E}, \mathcal{D}$ takes an additional input b from an associated *branch-set* $\mathcal{B}$. (For $\mathcal{E}$ it is given as the second input.) In particular, each $b^* \in \mathcal{B}$ yields particular algorithms $\mathcal{K}_{b^*}, \mathcal{E}_{b^*}, \mathcal{D}_{b^*}$. *If no branch input is specified, it is assumed to be a fixed "default" branch.* The following conditions must hold:

– (*One branch induces a hash.*) For any $b \in \mathcal{B}$, there is an induced hash function $\mathcal{H}_{\mathcal{E}_b} = (\mathcal{K}_b, H_{\mathcal{E}_b})$ with a 2^r-bounded hash range, where algorithm $\mathcal{K}_b$ returns pk_b, and $H_{\mathcal{E}_b}$ on inputs pk_b, x returns $\mathcal{E}(pk, b, x)$.
– (*Other branches encrypt.*) For any $b_1 \neq b_2 \in \mathcal{B}$, the triple $(\mathcal{K}_{b_1}, \mathcal{E}_{b_2}, \mathcal{D}_{b_2})$ is a deterministic encryption scheme.
– (*Hash branch is hidden.*) For any $b_1, b_2 \in \mathcal{B}$, any adversary A has negligible advantage, denoted $\mathbf{Adv}^{\mathrm{abo}}_{\mathcal{AE}}(A)$, in distinguishing the first outputs of $\mathcal{K}_{b_1}$ and $\mathcal{K}_{b_2}$.

In the case that for all $b \in \mathcal{B}$ the induced hash function $\mathcal{H}_{\mathcal{E}_b}$ in the first condition is universal, we say that scheme $\mathcal{AE}$ is *universal-ABO*.

The construction. For our general CCA-secure construction, we show how to adapt the IND-CCA *probabilistic* encryption scheme of [26] to the deterministic-encryption setting. In particular, our construction does not use a one-time signature scheme as in [26] but rather a TCR hash function.

Let $\mathcal{AE}_{\mathrm{hmm}} = (\mathcal{K}_{\mathrm{hmm}}, \mathcal{E}_{\mathrm{hmm}}, \mathcal{D}_{\mathrm{hmm}})$ be an ℓ-bit deterministic encryption scheme with a HHM and a $2^{r_{\mathrm{hmm}}}$-bounded hash range, let $\mathcal{AE}_{\mathrm{abo}} = (\mathcal{K}_{\mathrm{abo}}, \mathcal{E}_{\mathrm{abo}}, \mathcal{D}_{\mathrm{abo}})$ be an ℓ-bit ABO deterministic encryption scheme with branch set $\mathcal{B}$ and a $2^{r_{\mathrm{abo}}}$-bounded hash range, and let $\mathcal{H}_{\mathrm{tcr}} = (\mathcal{K}_{\mathrm{tcr}}, H_{\mathrm{tcr}})$ be a ℓ-bit hash function with a $2^{r_{\mathrm{tcr}}}$-bounded hash range $R \subseteq \mathcal{B}$. Key-generation algorithm $\mathcal{K}_{\mathrm{cca}}$ of the associated deterministic encryption scheme $\mathcal{AE}_{\mathrm{cca}} = (\mathcal{K}_{\mathrm{cca}}, \mathcal{E}_{\mathrm{cca}}, \mathcal{D}_{\mathrm{cca}})$ runs $\mathcal{K}_{\mathrm{tcr}}$, $\mathcal{K}_{\mathrm{hhm}}$, and $\mathcal{K}_{\mathrm{abo}}$ to obtain outputs $K_{\mathrm{tcr}}, (pk_{\mathrm{hhm}}, sk_{\mathrm{hhm}}), (pk_{\mathrm{abo}}, sk_{\mathrm{abo}})$, respectively; it then returns $(K_{\mathrm{tcr}}, pk_{\mathrm{hhm}}, pk_{\mathrm{abo}})$ as public key pk and sk_{hhm} as secret key sk. The encryption and decryption algorithms of are defined as follows:

Algorithm $\mathcal{E}_{\mathrm{cca}}(pk, m)$	**Algorithm** $\mathcal{D}_{\mathrm{cca}}(pk, sk, h\|c_1\|c_2)$
$h \leftarrow H_{\mathrm{tcr}}(K_{\mathrm{tcr}}, m)$	$m' \leftarrow \mathcal{D}_{\mathrm{hhm}}(sk_{\mathrm{hhm}}, c_1)$
$c_1 \leftarrow \mathcal{E}_{\mathrm{hhm}}(pk_{\mathrm{hhm}}, m)$	$c' \leftarrow \mathcal{E}_{\mathrm{cca}}(pk, m')$
$c_2 \leftarrow \mathcal{E}_{\mathrm{abo}}(pk_{\mathrm{abo}}, h, m)$	If $c' = h\|c_1\|c_2$ then return m'
Return $h\|c_1\|c_2$	Else return $\perp$

We show that if the HHM and ABO schemes in fact induce universal hash functions, and hash function $\mathcal{H}_{\mathrm{tcr}}$ is universal as well, then the construction indeed achieves PRIV-CCA-security for block-sources.

Theorem 4. *Let $\mathcal{AE}_{\mathrm{cca}} = (\mathcal{K}_{\mathrm{cca}}, \mathcal{E}_{\mathrm{cca}}, \mathcal{D}_{\mathrm{cca}})$ be as above, and suppose that $\mathcal{AE}_{\mathrm{hhm}}$ has a HUHB, $\mathcal{AE}_{\mathrm{abo}}$ is universal-ABO, and that $\mathcal{H}_{\mathrm{tcr}}$ is universal. Then for any adversary A, any (t, ℓ)-block-sources M_0, M_1, and all $\epsilon > 0$ such that $t \geq r_{\mathrm{tcr}} + r_{\mathrm{hhm}} + r_{\mathrm{abo}} + 2\log(1/\epsilon)$, there exists adversaries $B_{\mathrm{tcr}}, B_{\mathrm{hhm}}, B_{\mathrm{abo}}$ such that*

$$\mathbf{Adv}^{\text{priv1-ind-cca}}_{\mathcal{AE}_{\mathrm{cca}}}(A, M_0, M_1) \\ \leq\ 2 \cdot \left(\mathbf{Adv}^{\mathrm{tcr}}_{\mathcal{H}_{\mathrm{tcr}}}(B_{\mathrm{tcr}}) + \mathbf{Adv}^{\mathrm{hhm}}_{\mathcal{AE}_{\mathrm{hhm}}}(B_{\mathrm{hhm}}) + \mathbf{Adv}^{\mathrm{abo}}_{\mathcal{AE}_{\mathrm{abo}}}(B_{\mathrm{abo}}) + 3\epsilon \right) .$$

Furthermore, the running-times of $B_{\mathrm{tcr}}, B_{\mathrm{hhm}}, B_{\mathrm{abo}}$ are essentially that of A. □

The formal proof is in the full paper [7]. The idea is that, in the experiments for the PRIV1-IND-CCA adversary A, we may first replace the input branch to $\mathcal{AE}_{\text{abo}}$ by the hash (under $\mathcal{H}_{\text{TCR}}$) of "challenge message" m; then, using the secret key of $\mathcal{AE}_{\text{abo}}$ to answer A's decryption queries, we may replace $\mathcal{K}_{\text{hhb}}$ by the hash-inducing generator $\widetilde{\mathcal{K}}_{\text{hhb}}$. Crucial to this is that A cannot produce a valid decryption query that contains a hash h' colliding with the hash h of m, but this is guaranteed by the TCR property of $\mathcal{H}_{\text{tcr}}$ and the fact that each message has exactly one possible encryption. Now, the only information A sees on m are universal hashes of it. If m has enough min-entropy, then, intuitively, the LHL implies that each of these hashes are close to uniform, independent of the specific distribution of m, bounding A's advantage to be small.

One technical subtlety is that although the concatenation of independent instances of universal hash functions is again universal, in our case the universal hash function $\mathcal{H}_{\mathcal{E}_{\text{abo}}}$ coming from the ABO scheme depends (via the branch) on the outcome of the universal hash function $\mathcal{H}_{\text{tcr}}$. We overcome this by using the Generalized Leftover Hash Lemma and several observations from [18].

Application to witness-recovering decryption. We remark that our construction (as well as for that in Section 7), when converted into an IND-CCA probabilistic encryption scheme using the KEM-DEM-style conversion of [3],[9] yields, to the best of our knowledge, the first such scheme without ROs that is truly *witness-recovering*; that is, via the decryption process the receiver is able to recover *all* of the randomness used by a sender to encrypt the message. The constructs of [26] technically do not achieve this since, as the authors note, in their IND-CCA scheme the receiver does not recover the randomness used by the sender to generate a key-pair for the one-time signature scheme.

6 Schemes Based on DDH

In this section, we give instantiations of our general CPA- and CCA-secure constructions based the well-known decisional Diffie-Hellman assumption (DDH) in the corresponding groups. The "hidden branches" of the presented HHM and ABO schemes, as well as the TCR hash function in the instantiations are indeed universal, so Theorem 4 applies to show that they are PRIV-CCA-secure for block-sources. (Our CCA-secure construction uses the CPA one as a building-block, so we focus on instantiation of the former here.)

HHM and ABO schemes. In fact, the deterministic encryption scheme with HUHB and the universal-ABO deterministic encryption schemes are precisely the corresponding DDH-based constructs from [26] of lossy and ABO (with branch-set $\mathbb{Z}_p$ where prime p is the order of group $\mathbb{G}$ in which DDH holds) trapdoor functions with 2^k-bounded hash ranges, where k is the bit-size of p. It suffices to observe that the "lossy branches" of these functions are in fact universal. These constructs are recalled in Appendix A, where this observation is justified. Our results demonstrate that these constructs have stronger security properties than were previously known.

[9] Note that security of the resulting probabilistic scheme only requires the base deterministic scheme to be secure for the encryption of a single high-entropy message.

UNIVERSAL-TCR HASH. To fully instantiate our CCA-secure construction, it remains to design an ℓ-bit hash function whose range is contained in the branch-set $\mathbb{Z}_p$ of the DDH-based ABO scheme, and which is both universal and TCR (we will call such hashes "universal-TCR"). We accomplish this slightly differently for two popular choices of group $\mathbb{G}$ in which DDH is believed to hold, giving rise to two possible concrete instantiations of the construction.

Instantiation 1. Let $\mathbb{G}$ be a group of prime order $p = 2q+1$, where q is also prime (i.e. p is a so-called *safe* prime). Let p have size k. This covers the case of $\mathbb{G}$ as an appropriate elliptic-curve group where DDH is hard. Let $QR(\mathbb{Z}_p^*) = \{x^2 \mid x \in \mathbb{Z}_p^*\}$ be the subgroup of quadratic residues modulo p. Note that $QR(\mathbb{Z}_p^*)$ has order $(p-1)/2 = q$. (Also note that we can sample from $QR(\mathbb{Z}_p^*)$ by choosing a random $x \in \mathbb{Z}_p^*$ and returning x^2.) In this case we can use the following hash function, based on the general construct from [11]. Define the key-generation and hash algorithms of ℓ-bit hash function $\mathcal{H}_1 = (\mathcal{K}_1, H_1)$ as follows:

Algorithm $\mathcal{K}_1$	**Algorithm** $H_1((R_1, \ldots, R_l), x)$
$R_1, \ldots, R_l \xleftarrow{\$} QR(\mathbb{Z}_p^*)$	$\pi \leftarrow \prod_{i=1}^{l} R_i^{x_i}$
Return $(R_1, \ldots, R_l)$	Return π

Above, x_i denotes the i-th bit of string x.

Proposition 1. *Hash function $\mathcal{H}_1$ defined above is CR assuming the discrete-logarithm problem (DLP) is hard in $QR(\mathbb{Z}_p^*)$, and is universal with a 2^{k-2}-bounded hash range $QR(\mathbb{Z}_p^*)$ contained in $\mathbb{Z}_p$.*

The proof is in the full paper [7]. Note that the hardness of the DLP is a weaker assumption than DDH (although it is made in a different group).

Instantiation 2. Now let $\mathbb{G}$ be $QR(\mathbb{Z}_{p'}^*)$, where $p' = 2p + 1$ is as before a safe prime, so that $|\mathbb{G}| = p$ is also prime. This is another popular class of groups where DDH is believed hard. To instantiate the universal-TCR hash function in this case, we would like to use $\mathcal{H}_1$ from Instantiation 1, but this cannot (immediately) work since $QR(\mathbb{Z}_{p'}^*)$ is not a subset of $\mathbb{Z}_p$. However, we can modify hash algorithm H_1 to output $\mathsf{encode}(\pi)$ instead of π, where encode is an bijection from $QR(\mathbb{Z}_{p'}^*)$ to $\mathbb{Z}_p$. Namely, we can use the "square-root coding" function from [12]: $\mathsf{encode}(\pi) = \min\{\pm\pi^{(p'+1)/4}\}$. Here $\pm\pi^{(p'+1)/4}$ are the two square-roots of π, using the fact that for any safe prime $p' > 5$ we have p' is congruent to 3 mod 4.

While our DDH-based schemes are a definite proof of concept that secure deterministic encryption can be constructed from a widely-accepted number-theoretic assumption, they are rather inefficient. In particular, the constructs of [26] follow a "matrix encryption" approach and have public keys with order ℓ^2 group elements and ciphertexts with order ℓ group elements. We seek more efficient schemes. However, the other instantiations of lossy and ABO TDFs given in [26] do not (immediately) give universal lossy branches. To overcome this difficulty, we first show how to extend our general constructions in an efficient way to provide security when *any* lossy and ABO TDFs are used.

7 Extended General Constructions

Generalized "Crooked" LHL. In our security proofs we used the fact that the "lossy modes" of the underlying primitives, unlike those defined in [26], act as universal hash functions, allowing us to apply the LHL. However, the conclusion of the LHL was actually stronger than we needed, telling us that output of the lossy modes are *uniform* (and not merely input-independent). We show that the extra universality requirement can actually be avoided, not only for the HHB and ABO schemes but also the TCR hash function, by slightly extending our constructions. The extensions derive from a variant of the LHL due to Dodis and Smith [19, Lemma 12]. We actually need the following generalization of it analogous to the generalization of the standard LHL in [18].

Lemma 1. (Generalized "Crooked" LHL) *Let $\mathcal{H} = (\mathcal{K}, H)$ be an ℓ-bit pairwise-independent hash function with range R, and let $f : R \to S$ be a function to a set S. Let the random variable K describe the key generated by $\mathcal{K}$, and U the uniform distribution over R. Then for any random variable X over $\{0,1\}^\ell$ and any random variable Z such that $\tilde{\mathrm{H}}_\infty(X|Z) \geq \log|S| + 2\log(1/\epsilon) - 2$, we have $\Delta\big((f(H(K,X)), Z, K), (f(U), Z, K)\big) \leq \epsilon$.*

The proof is the full version [7]. Intuitively, the lemma says that if we compose a pairwise-independent hash function with *any* lossy function, the output of the composition is essentially input-independent (but not necessarily uniform), as long as the input has enough (average conditional) min-entropy. This suggests the following extension to our CCA-secure construction. (We treat the CCA case here, since the extension to our CPA-secure construction is evident from it.)

Extended CCA-secure construction. Let $\mathcal{E}_{\rm cca} = (\mathcal{K}_{\rm cca}, \mathcal{E}_{\rm cca}, \mathcal{D}_{\rm cca})$ be as defined in Section 5.2, and let $\mathcal{H}_{\rm pi} = (\mathcal{K}_{\rm pi}, H_{\rm pi})$ be an ℓ-bit invertible pairwise-independent hash function with range $\{0,1\}^\ell$. Invertibility of $\mathcal{H}_{\rm pi}$ means that there is a polynomial-time algorithm I such that for all $K_{\rm pi}$ that can be output by $\mathcal{K}_{\rm pi}$ and all $m \in \{0,1\}^\ell$ we have $I(\mathcal{K}_{\rm pi}, H_{\rm pi}(\mathcal{K}_{\rm pi}, m))$ outputs m. The key-generation algorithm $\mathcal{K}^+_{\rm cca}$ of the associated composite scheme $\mathcal{AE}^+_{\rm cca} = (\mathcal{K}^+_{\rm cca}, \mathcal{E}^+_{\rm cca}, \mathcal{D}^+_{\rm cca})$ is the same as $\mathcal{K}_{\rm cca}$ except it also generates three independent hash keys $K_{\rm pi,1}, K_{\rm pi,2}, K_{\rm pi,3}$ via $\mathcal{K}_{\rm pi}$ which are included in the public key pk. The encryption and decryption algorithms are defined as follows:

Alg $\mathcal{E}^+_{\rm cca}((\{K_{\rm pi,\ i}\}, pk_{\mathcal{AE}}), m)$	**Alg** $\mathcal{D}^+_{\rm cca}((\{K_{\rm pi,\ i}\}, pk_{\mathcal{AE}}), sk_{\mathcal{AE}}, c)$
For $i = 1$ to 3 do $h_i \leftarrow H_{\rm pi}(K_{{\rm pi},i}, m)$	Parse c as $h \\| c_1 \\| c_2$
$h \leftarrow H(K_{\rm tcr}, h_1)$	$h'_1 \leftarrow \mathcal{D}_{\rm hhm}(sk_{\rm hhm}, c_1)$
$c_1 \leftarrow \mathcal{E}_{\rm hhm}(pk_{\rm hhm}, h_2)$	$m' \leftarrow I(K_{\rm pi,2}, h'_1)$
$c_2 \leftarrow \mathcal{E}_{\rm abo}(pk_{\rm abo}, h_3)$	$c' \leftarrow \mathcal{E}^+_{\rm cca}((\{K_{\rm pi,\ i}\}, pk_{\mathcal{AE}}), m')$
Return $h \\| c_1 \\| c_2$	If $c' = h \\| c_1 \\| c_2$ then return m'
	Else return $\perp$

Concretely, viewing ℓ-bit strings as elements of the finite field $\mathbb{F}_{2^\ell}$, we can use for $\mathcal{H}_{\rm pi}$ the standard construct $\mathcal{H}_\ell = (\mathcal{K}_\ell, H_\ell)$ where $\mathcal{K}_\ell$ outputs a random

$a, b \in \mathbb{F}_{2^\ell}$ and H_ℓ on inputs $(a, b), x$ returns $ax + b$, which is clearly invertible.[10] Using Lemma 1, we obtain the following result.

Theorem 5. *Let $\mathcal{AE}^+_{\rm cca} = (\mathcal{K}^+_{\rm cca}, \mathcal{E}^+_{\rm cca}, \mathcal{D}^+_{\rm cca})$ be as defined above. Then for any adversary A, any (t, ℓ)-block-sources M_0, M_1 and any $\epsilon > 0$ such that $t \geq r_{\rm tcr} + r_{\rm hhm} + r_{\rm abo} + 2\log(1/\epsilon) + 2$, there exist adversaries $B_{\rm tcr}, B_{\rm hhm}, B_{\rm abo}$ such that*

$$\begin{aligned} &\mathbf{Adv}^{\text{priv1-ind-cca}}_{\mathcal{AE}^+_{\rm cca}}(A, M_0, M_1) \\ &\quad \leq\ 2 \cdot \left(\mathbf{Adv}^{\rm tcr}_{\mathcal{H}_{\rm tcr}}(B_{\rm tcr}) + \mathbf{Adv}^{\rm hhm}_{\mathcal{AE}_{\rm hhm}}(B_{\rm hhm}) + \mathbf{Adv}^{\rm abo}_{\mathcal{AE}_{\rm abo}}(B_{\rm abo}) + 3\epsilon \right) . \end{aligned}$$

Furthermore, the running-times of $B_{\rm tcr}, B_{\rm hhm}, B_{\rm abo}$ are essentially that of A. $\square$

Decreasing the key size. A potential drawback of the extended CCA-secure scheme is its public-key size, due to including three hash keys for $\mathcal{H}_{\rm pi}$. But in fact we can usually re-use the same key, i.e. take $K_{\rm pi,1} = K_{\rm pi,2} = K_{\rm pi,3}$. For the security proof to go through, we just need the minor technical condition that the range of the hash function $\mathcal{H}_{\mathcal{E}_{\rm abo}}$[11] induced by the lossy branch of the ABO scheme be independent of the particular branch b.This condition is met by all known instantiations. In the proof, this condition allows us to apply Lemma 1 to the "concatenation" of the hash functions $\mathcal{H}_{\rm tcr}$, $\mathcal{H}_{\rm hhb}$ and $\mathcal{H}_{\rm abo}$. Details are provided in the full paper [7].

Advantages. In our extended CCA-secure construction, we can use *any* lossy and ABO TDF, as defined in [26]. In particular, we can use the Paillier- and lattice-based constructs of [26], although we obtain even more efficient Paillier-based schemes in the next section. Also, since $\mathcal{H}_{\rm tcr}$ in the extended scheme need only be TCR and "lossy," it can be a cryptographic hash function such as SHA256 or the efficient TCR constructs of [5, 29] in practice. (Security of the basic scheme required $\mathcal{H}_{\rm tcr}$ to be both TCR and *universal*, whereas cryptographic hash functions fail to meet the latter.) Such instantiation of $\mathcal{H}_{\rm tcr}$ is compatible with any ABO scheme with branch-set $\mathcal{B}$ for which $\{0, 1\}^n \subseteq \mathcal{B}$ for n sufficiently large (so that the probability of hashing to the "default" lossy branch of the ABO scheme is small). Again, this is satisfied for all known instantiations.

8 Efficient Schemes Based on Paillier's DCR Assumption

To improve on efficiency over their DDH-based constructs, [26] suggests basing their matrix-encryption approach on Paillier encryption [25] (which uses the group $\mathbb{Z}_{N^2}$ for an RSA modulus N) instead. One then obtains HHM (or lossy

[10] Note that $\mathcal{K}_\ell$ must compute a representation of $\mathbb{F}_{2^\ell}$, which can be done in expected polynomial-time. Alternatively, a less-efficient, matrix-based instantiation of $\mathcal{H}_{\rm pi}$ runs in strict polynomial time and is invertible with high probability (over the choice of the hash key).

[11] Technically, $\mathcal{H}_{\mathcal{E}_{\rm abo}}$ should have a third subscript "b," but we drop it here and in similar instances for ease of notation.

TDF) and ABO schemes with an N-bounded hash range and offering roughly a factor $\log N$ savings in public-key and ciphertext size, namely public keys contain order $(\ell/\log N)^2$ group elements and ciphertexts order $\ell/\log N$ group elements for ℓ-bit messages, and for which encryption requires the latter amount of exponentiations. Based on a technique introduced by Damgård and Nielsen [14, 15], we propose new Paillier-based schemes that use an entirely different (i.e. non-matrix-encryption-based) approach and have even better efficiency: they are essentially length-preserving, have about ℓ-bit public keys, and compare to standard Paillier encryption computationally.

SETTING. Let $\mathcal{K}$ be an algorithm that outputs $(N, (p, q))$ where $N = pq$ and p, q are random $k/2$-bit primes. Paillier's *decisional composite residuosity* (DCR) *assumption* [25] states that any poly-time adversary A has negligible advantage in distinguishing a from $a^N \bmod N^2$ for random $(N, (p, q))$ output by $\mathcal{K}$ and random $a \in \mathbb{Z}^*_{N^2}$. Let $s \geq 1$ be polynomial in k. Our schemes actually use a generalization of Paillier encryption, based on the same assumption, to the group $\mathbb{Z}_{N^{s+1}}$ due to Damgård and Jurik [13], with some modifications in the spirit of [14, 15]. The schemes have message-space $\{0,1\}^{(s+1)(k-1)}$ (i.e. $\ell = (s+1)(k-1)$), where we regard messages as elements of $\{0, \ldots, 2^{s(k-1)}\} \times \{1, \ldots, 2^{k-1}+1\}$, chosen so that it is contained in the "usual" message-space $\mathbb{Z}_{N^s} \times \mathbb{Z}_N$ for any possible N output by $\mathcal{K}$.

THE NEW DETERMINISTIC ENCRYPTION SCHEME WITH HHM. Define scheme $\mathcal{AE}_{\mathrm{hhm}} = (\mathcal{K}_{\mathrm{hhm}}, \tilde{\mathcal{K}}_{\mathrm{hhm}}, \mathcal{E}_{\mathrm{hhm}}, \mathcal{D}_{\mathrm{hhm}})$ as follows (decryption is specified below):

Alg $\mathcal{K}_{\mathrm{hhm}}$	**Alg** $\tilde{\mathcal{K}}_{\mathrm{hhm}}$	**Alg** $\mathcal{E}_{\mathrm{hhm}}((g, N), (x, y))$
$(N, (p, q)) \xleftarrow{\$} \mathcal{K}$	$(N, (p, q)) \xleftarrow{\$} \mathcal{K}$	If $\gcd(y, N) \neq 1$
$a \xleftarrow{\$} \mathbb{Z}^*_N$	$a \xleftarrow{\$} \mathbb{Z}^*_N$	Then abort
$g \leftarrow (1+N)a^{N^s} \bmod N^{s+1}$	$\tilde{g} \leftarrow a^{N^s} \bmod N^{s+1}$	$c \leftarrow g^x y^{N^s} \bmod N^{s+1}$
Return $((g, N), (p, q))$	Return $(\tilde{g}, N)$	Return c

Decryption algorithm $\mathcal{D}_{\mathrm{hhm}}$ on inputs $(g, N), (p, q), c$ uses standard Paillier-decryption as in [13] to recover x, then computes y by taking the N^s-th root of c/g^x (which can be done efficiently given p, q) and returns (x, y). The fact that the scheme indeed has a HHM, i.e. that the first outputs of $\mathcal{K}_{\mathrm{hhm}}$ and $\tilde{\mathcal{K}}_{\mathrm{hhm}}$ above are indistinguishable, follows under DCR by security of the underlying "randomized" encryption scheme of [13]: g output by $\mathcal{K}_{\mathrm{hhm}}$ is an encryption of 1 under this scheme and $\tilde{g}$ output by $\tilde{\mathcal{K}}_{\mathrm{hhm}}$ is an encryption of 0.

Note that the hash range is isomorphic to $\mathbb{Z}^*_N$, hence the scheme has a 2^k-bounded hash range. Also note that the size of this range does *not* depend on parameter s; in hidden hash mode the encryption function "looses" a $1-1/(s+1)$ fraction of the information on the plaintext, so by increasing s we can make the scheme arbitrarily (i.e. $1 - o(1)$) lossy as defined in [26]. This has some useful consequences. First, it allows to securely encrypt long messages with small min-entropy relative to the length of the message. Second, it permits a purely black-box construction of an ABO scheme with many branches having the same amount of lossiness, via the reduction in [26, Section 3.3]. (The latter applies in the lossy TDF context as well.) However, we obtain a much more efficient ABO scheme directly in the following.

THE NEW ABO DETERMINISTIC ENCRYPTION SCHEME. Define scheme $\mathcal{AE}_{\mathrm{abo}} = (\mathcal{K}_{\mathrm{abo}}, \mathcal{E}_{\mathrm{abo}}, \mathcal{D}_{\mathrm{abo}})$ with branch-set Z_{N^s} as follows:

Algorithm $\mathcal{K}_{\mathrm{abo}}(b^*)$	**Algorithm** $\mathcal{E}_{\mathrm{abo}}((g,N), b, (x,y))$
$(N,(p,q)) \xleftarrow{\$} \mathcal{K}$	If $\gcd(y,N) \neq 1$ then abort
$a \xleftarrow{\$} \mathbb{Z}_N^*$	$h \leftarrow g/(1+N)^b \bmod N^{s+1}$
$g \leftarrow (1+N)^{b^*} a^{N^s} \bmod N^{s+1}$	Else $c \leftarrow h^x y^{N^s} \bmod N^{s+1}$
Return $((g,N),(p,q))$	Return c

where decryption works essentially as in the previous scheme. A similar analysis shows that under DCR it is indeed an ABO scheme with 2^k-bounded hash range.

TCR HASH. To instantiate our extended CCA-secure construction, it remains to specify a TCR hash function with range the branch-set Z_{N^s} of the above ABO scheme. One way is to use a "heuristic" cryptographic hash function, as discussed in Section 7. This approach also yields, via the KEM-DEM-style conversion of [3], a quite efficient, witness-recovering IND-CCA (probabilistic) encryption scheme. However, for completeness, we give below an alternative construction of a provably CR hash function based on the computational analogue of DCR, which dovetails nicely with our ABO scheme for $s \geq 2$.

We now regard the $(s+1)(k-1)$-bit messages as elements of $(x_1, \ldots, x_s, y) \in \{0, \ldots, 2^{k-1}\}^s \times \{1, \ldots, 2^{k-1}+1\}$ and define hash function $\mathcal{H}_2 = (\mathcal{K}_2, H_2)$ as:

Algorithm $\mathcal{K}_2$	**Algorithm** $H_2((g_1, \ldots, g_s), (x_1, \ldots x_s, y))$
$(N,(p,q)) \xleftarrow{\$} \mathcal{K}$	$\pi \leftarrow \ g_1^{x_1} \cdots g_s^{x_s} y^N \bmod N^2$
For $i = 1$ to s do:	Return π
$\quad a_i \xleftarrow{\$} Z_N^*$; $g_i \leftarrow a_i^N \bmod N^2$	
Return $(g_1, \ldots, g_s)$	

Proposition 2. *Hash function $\mathcal{H}_2$ defined above is CR assuming the computational composite residuosity assumption [25] holds (relative to $\mathcal{K}$).*

Proof. Given an adversary A that produces a collision, we construct an adversary A' which computes an N-th root of a random N-th power $h = a^N$ in $\mathbb{Z}_{N^2}$. On input h, A' chooses a random index $i^* \in \{1, \ldots, s\}$ and runs $\mathcal{K}_2$ but replaces g_{i^*} by $g_{i^*} \leftarrow h$. Then, it runs A on inpt $(g_1, \ldots, g_s, y)$ and obtains a collision with probability $\mathbf{Adv}^{\mathrm{CR}}(A)$, i.e., $(x_1, \ldots, x_s, y) \neq (x_1, \ldots, x_s, y)$ such that $g_1^{x_1} \cdots g_s^{x_s} y^N = g_1^{x_1'} \cdots g_s^{x_s'} {y'}^N$. In this case, note that $x_j \neq x_j'$ for some j, as otherwise $y^N = {y'}^N$ modulo N^2 which implies that also $y = y'$ modulo N, and with probability $1/s$: $i^* = j$. Furthermore, note that we may assume that $x_{i^*} - x_{i^*}'$ is co-prime with N, as otherwise A' can immediately factor N. It follows that if indeed $i^* = j$ then A' can efficiently compute integers σ and τ such that $2\sigma(x_{i^*} - x_{i^*}') + \tau N = 1$. Raising both sides of

$$g_{i^*}^{x_{i^*} - x_{i^*}'} = \prod_{\substack{i=1 \\ i \neq i^*}}^{s} g_i^{x_i' - x_i} \cdot \left(\frac{y'}{y}\right)^N$$

to the power σ and multiplying both sides with $g^{\tau N}$ results in

$$g_{i^*} = \Bigg(\prod_{\substack{i=1 \\ i \neq i^*}}^{s} a_i^{x'_i - x_i} \cdot \frac{y'}{y} g^{\tau} \Bigg)^N .$$

Thus, with probability $\mathbf{Adv}^{\mathrm{CR}}(A)/s$, A' obtaines a N-th root of $g_{i^*} = h$. $\square$

Note that the hash function is "compatible" with the above ABO deterministic encryption scheme in that a hash value it produces lies in Z_{N^s} as long as $s \geq 2$ and the N from the hash function is not larger than the N from the ABO scheme; in fact, it is not too hard to verify that the hash function and the ABO scheme may safely use the same N, so that the latter condition is trivially satisfied.

Acknowledgements

We thank Eike Kiltz, Chris Peikert, and Yevgeniy Dodis for helpful discussions and comments, and the anonymous reviewers of Crypto 2008 for their suggestions. Alexandra Boldyreva was supported in part by NSF CAREER award 0545659. Serge Fehr was supported by a VENI grant from the Dutch Organization for Scientific Research (NWO). Adam O'Neill was supported in part by the above-mentioned grant of the first author.

References

1. Bellare M., Boldyreva A., O'Neill A.: Deterministic and efficiently searchable encryption. In: CRYPTO 2007. LNCS, vol. 4622. Springer (2007)
2. Bellare M., Boldyreva A., Palacio A.: An uninstantiable random-oracle-model scheme for a hybrid-encryption problem. In: EUROCRYPT 2004. LNCS vol. 3027. Springer (2004)
3. Bellare M., Fischlin M., O'Neill A., Ristenpart T.: Deterministic encryption: Definitional equivalences and constructions without random oracles. In: CRYPTO 2008. LNCS. Springer (2008)
4. Bellare M., Rogaway P.: Random oracles are practical: a paradigm for designing efficient protocols. In: CCS 1993. ACM (1993)
5. Bellare M., Rogaway P.: Collision-resistant hashing: Towards making UOWHFs practical. In: CRYPTO 1997. LNCS vol. 1294. Springer (1997)
6. Bennett C., Brassard G., Crepeau C., Maurer U.: Generalized privacy amplification. In: Transactions on Information Theory. 41(6), IEEE (1995)
7. Boldyreva A., Fehr S. and O'Neill A.: On notions of security for deterministic encryption, and efficient constructions without random oracles. Full version of this paper. Available at `http://eprint.iacr.org/2008/` (2008)
8. Canetti R., Goldreich O., Halevi S.: The random oracle methodology, revisited. In: STOC 1998. ACM (1998)
9. Carter J. L., Wegman M. N.: Universal classes of hash functions. In: Journal of Computer and System Sciences, vol. 18 (1979)
10. Carter J. L., and Wegman M. N.: New hash functions and their use in authentication and set equality. In: Journal of Computer and System Sciences, vol. 22 (1981)

11. Chaum, D., van Heijst E., Pfitzmann B.: Cryptographically strong undeniable signatures, unconditionally secure for the signer. In: CRYPTO 1991. LNCS vol. 576. Springer (1992)
12. Cramer R., Shoup V.: A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In: CRYPTO 1998, LNCS, vol. 1462. Springer (1998)
13. Damgård I, Jurik M.: A generalisation, a simplification and some applications of paillier's probabilistic public-key system. In: *PKC 2001*, LNCS, vol. 1992. Springer (2001)
14. Damgård I., Nielsen J.-B.: Perfect hiding and perfect binding universally composable commitment schemes with constant expansion factor. In: CRYPTO 2002, LNCS, vol. 2442. Springer (2002)
15. Damgård I., Nielsen J.-B.: Universally composable efficient multiparty computation from threshold homomorphic encryption. In: CRYPTO 2003, LNCS, vol. 2729. Springer (2003)
16. Desrosiers S.: Entropic security in quantum cryptography. ArXiv e-Print quant-ph/0703046, `http://arxiv.org/abs/quant-ph/0703046` (2007)
17. Desrosiers S. and Dupuis F.: Quantum entropic security and approximate quantum encryption. arXiv e-Print quant-ph/0707.0691, `http://arxiv.org/abs/0707.0691` (2007)
18. Dodis Y., Ostrovsky R., Reyzin L., Smith A: Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. Available from `http://eprint.iacr.org/2003/235`. Preliminary version appeared in: *EUROCRYPT 2004*. LNCS, vol. 3027. Springer (2004)
19. Dodis Y., Smith A: Correcting errors without leaking partial information. In: STOC 2005. ACM (2005)
20. Dodis Y., Smith A: Entropic security and the encryption of high entropy messages. In: TCC 2005. LNCS, vol. 3378. Springer (2005)
21. ElGamal T.: A public key cryptosystem and signature scheme based on discrete logarithms. In: Transactions on Information Theory, vol. 31. IEEE (1985)
22. Goldwasser S., Tauman Kalai Y.: On the (in)security of the Fiat-Shamir paradigm. In: *FOCS 2003*. IEEE 2003.
23. Hastad J., Impagliazzo R., Levin L., Luby M.: A pseudorandom generator from any one-way function. In: *Journal of Computing* 28(4). SIAM (1999)
24. Naor M., Yung M.: Universal one-way hash functions and their cryptographic applications. In: *STOC 1989*. ACM (1989)
25. Paillier P.: Public-key cryptosystems based on composite degree residuosity classes. In: *EUROCRYPT 1999*, LNCS, vol. 1592. Springer (1999).
26. Peikert C, Waters B.: Lossy trapdoor functions and their applications. In: *STOC 2008*. ACM 2008.
27. Rosen A., Segev G.: Efficient lossy trapdoor functions based on the composite residuosity assumption. In: Cryptology ePrint Archive: Report 2008/134, (2008).
28. Russell A., Wang H.: How to fool an unbounded adversary with a short key. In: *EUROCRYPT 2002*, LNCS, vol. 2332. Springer (2002).
29. Shoup V.: A composition theorem for universal one-way hash functions. In: *EUROCRYPT 2000*, LNCS, vol. 1807. Springer (2000)

A DDH-Based Lossy and ABO TDFs of Peikert-Waters

In [26] the authors introduce a form of "matrix encryption" that they use to realize lossy and ABO TDFs based on encryption schemes allowing some linearalgebraic operations to be performed on ciphertexts. We briefly recall this and the resulting schemes here (using our terminology of HHM and ABO deterministic encryption schemes rather than lossy and ABO TDFs). For concreteness we describe the schemes based on DDH. Moreover, although this was not shown in [26], the "lossy branches" of the DDH-based schemes are *universal*, so we can use them towards instantiating our basic CPA- and CCA-secure constructions. Throughout the description we fix a group $\mathbb{G}$ of prime order p with generator g in which DDH is believed to hold.

ElGamal-based matrix encryption. We first review the ElGamal-based method of [26] for encrypting $\ell \times \ell$ boolean matrices. The public key is $(g^{s_1}, \ldots, g^{s_\ell})$, where $s_1, \ldots, s_\ell \in \mathbb{Z}_p$ are random, and $(s_1, \ldots s_\ell)$ is the secret key. The encryption of an $\ell \times \ell$ boolean matrix $A = (a_{ij})$ is the matrix $C = (c_{ij})$ of pairs of elements in $\mathbb{G}$, where $c_{ij} = (g^{a_{ij}} g^{s_i \cdot r_i}, g^{r_i})$ for random $r_1, \ldots, r_\ell \in \mathbb{Z}_p$. Note that the same randomness is re-used for elements in the same row and the same component of the public key is re-used for elements in the same column. Under the DDH assumption, the encryption of any matrix using this scheme is indistinguishable from the encryption of any other one [26, Lemma 5.1].

The schemes. We briefly describe the DDH-based deterministic encryption scheme with HHM from [26]. The (normal) key-generation algorithm of the scheme outputs an encryption of the $(\ell \times \ell)$ identity-matrix I under the above scheme as the public key, and the s_j's as the secret key. To encrypt a message $\boldsymbol{x} = (x_1, \ldots, x_\ell) \in \{0,1\}^\ell$ one multiplies $\boldsymbol{x}$ (from the left) into the encrypted public-key matrix by using the homomorphic property of ElGamal: ciphertext $\boldsymbol{c} = (c_1, \ldots, c_\ell)$ is computed as

$$c_j = \left(\prod_i u_{ij}^{x_i}, \prod_i v_{ij}^{x_i} \right).$$

It is easy to verify that $c_j = \left(g^\rho, g^{x_j} h_j^\rho\right)$ with $\rho = \sum_i r_i x_i \in \mathbb{Z}_p$, so that standard ElGamal decryption allows to recover x_j when given s_j (using the fact that $x_j \in \{0,1\}$). The alternate key-generation algorithm of the scheme outputs an encryption of the $(\ell \times \ell)$ all-zero matrix rather than of the identity matrix, so that the encryption of a message $\boldsymbol{x}$ results in the ciphertext $\boldsymbol{c}$ with $c_j = \left(g^\rho, h_j^\rho\right)$ where, as before, $\rho = \sum_i r_i x_i$. Thus, $\boldsymbol{c}$ only contains limited information on $\boldsymbol{x}$, namely $\rho = \sum_i r_i x_i \in \mathbb{Z}_p$. This makes the encryption function *lossy*, as required in [26], but it is also easy to see that it also makes the encryption function a universal hash function. Indeed, the encryptions $\boldsymbol{c}$ and $\boldsymbol{c}'$ of two distinct messages $\boldsymbol{x}$ and $\boldsymbol{x}'$ collide if and only if the corresponding $\rho = \sum_i r_i x_i$ and $\rho' = \sum_i r_i x'_i$ collide, which happens with probability $1/q$ (over the choices of the r_i's). Thus, for any ℓ, we obtain an ℓ-bit deterministic encryption scheme with HHM having 2^k-bounded hash range, where k is the bit-size of p. We omit the description of the corresponding DDH-based ℓ-bit ABO scheme with 2^k-bounded hash range obtained from [26]. Essentially the same analysis applies to show that its lossy branch is universal as well.