Reproducible Circularly-Secure Bit Encryption: Applications and Realizations

Mohammad Hajiabadi and Bruce M. Kapron(✉)

Department of Computer Science, University of Victoria, Victoria V8W 3P6, Canada
{mhaji,bmkapron}@cs.uvic.ca

Abstract. We give generic constructions of several fundamental cryptographic primitives based on a new encryption primitive that combines *circular security* for bit encryption with the so-called *reproducibility property* (Bellare et al. PKC 2003). At the heart of our constructions is a novel technique which gives a way of de-randomizing reproducible public-key bit-encryption schemes and also a way of reducing one-wayness conditions of a constructed trapdoor-function family (TDF) to circular security of the base scheme. The main primitives that we build from our encryption primitive include *k-wise one-way* TDFs (Rosen and Segev TCC 2009), CCA2-secure encryption and deterministic encryption. Our results demonstrate a new set of applications of circularly-secure encryption beyond fully-homomorphic encryption and symbolic soundness. Finally, we show the plausibility of our assumptions by showing that the DDH-based circularly-secure scheme of Boneh et al. (Crypto 2008) and the subgroup indistinguishability based scheme of Brakerski and Goldwasser (Crypto 2010) are both reproducible.

Keywords: Circular security · Correlated-input security · Trapdoor functions · (non-)shielding CCA construction · Deterministic encryption

1 Introduction

A central problem in cryptography is delineating the assumptions required for the existence of cryptographic primitives. One way to differentiate assumptions is by whether they refer to the hardness of a *specific* computational problem (e.g., factoring), or refer *generically* to a class of problems (e.g., inverting efficiently computable functions). Assumptions of the former sort often lead to primitives which are more practical, e.g., in terms of efficiency or levels of security achieved. Those of the latter sort are useful for gaining deeper insights into the security requirements of a primitive, and also as a means of unifying specific assumptions. However, these approaches are not mutually exclusive. In particular, in cases where we have not been able to obtain constructions based on generic assumptions, we may consider strengthening an assumption with some more specific properties. This is the approach we take in this paper. By adding a syntactic property to *circularly-secure* bit encryption, we are able to obtain constructions of several powerful cryptographic primitives.

R. Gennaro and M. Robshaw (Eds.): CRYPTO 2015, Part I, LNCS 9215, pp. 224–243, 2015.
DOI: 10.1007/978-3-662-47989-6_11

More precisely, we give constructions of various cryptographic primitives based on a general encryption primitive, which combines *circular security* with a property called *reproducibility* [5], which, the latter, gives a way of reusing randomness across independent public keys. We show the following results.

(1) We give a novel generic construction of TDFs from reproducible bit encryption, and under this construction we show that successively stronger circular-security conditions result in successively stronger one-wayness conditions: we give a hierarchy of circular security notions, called *k-rec circular security*, all of which are weaker than those of [2,11,12], and show if the base scheme is k-rec circularly secure, the constructed TDF is k-wise one-way, in the sense of [28].
(2) We show how to extract many hardcore bits for our constructed TDFs, and by applying the results of [28] we obtain a blackbox (BB) construction of CCA2-secure encryption from our assumptions. Our CCA2 construction is *non-shielding* in the sense of [18]. We partially justify this fact by showing wrt a weaker encryption primitive than ours, a non-shielding BB CCA2 construction is possible, while a shielding CCA2 construction is BB impossible.
(3) By slightly extending our base primitive, we show how to obtain deterministic encryption schemes secure under *block-source* inputs, as defined by [9].
(4) We realize our base encryption primitive by showing the circularly-secure schemes of [11,12] are reproducible.

In what follows, we provide some background, give a more detailed exposition of our results and describe our constructions and proof techniques. First of all, we assume the following notation and conventions throughout the introduction. Unless otherwise stated, an encryption scheme is bit encryption with randomness space $\{0,1\}^{\rho}$ and secret-key space $\{0,1\}^{l}$, where $l = l(n)$ and $\rho = \rho(n)$; by $E_{pk}(m)$, for $m \in \{0,1\}^{*}$, we mean bitwise encryption of m.

Trapdoor Functions. Central to public-key cryptography is the notion of *injective trapdoor one-way function*, which refers to a family of functions, where each function in the family is easy to compute, but a randomly chosen function is hard to invert without a *trapdoor key*. A related notion is *witness-recovering CPA-secure encryption*: CPA-secure public-key encryption (PKE) where the decryption algorithm also recovers the randomness used for encryption. It is well-known that these two primitives are equivalent. However, as shown by Gertner et al. [19], there is a BB separation between CPA-secure PKE and TDFs. An interpretation of this result is that a construction of a TDF from PKE should either be non-blackbox, or should rely on specific properties of the PKE. Indeed, under specific assumptions, TDFs may be constructed "directly" (e.g., under the factoring assumption), or may be constructed by using the specifics of a particular PKE scheme (e.g., the strong homomorphisms, among other properties, of ElGamal encryption [26]).

A folklore attempt to build a TDF from PKE is to encrypt a message x under a randomness string derived deterministically from x. However, by [19], such a methodology is in general not sound. A naturally arising question is what properties of PKE enable sound realizations of this approach. The starting

point of our work is a related question, namely: when does a PKE scheme allow "secure" encryption of r, using r itself as randomness? By security we mean it be hard to recover r from $(E_{pk_1}(r_1; r), \ldots, E_{pk_\rho}(r_\rho; r))$. Note that this immediately yields a TDF.

To address this question we first review a property of PKE schemes, called *reproducibility* [5]: $\mathcal{E} = (Gen, E, D)$ is reproducible if there exists an efficient deterministic function R, which given a ciphertext $c = E_{pk}(m; r)$, a message m_1, and public/secret keys (pk_1, sk_1), computes $E_{pk_1}(m_1; r)$, which we denote by $R(c, m_1, sk_1)$. Namely, there is an efficient way to transfer the randomness underlying a given encryption to another, provided the secret key for the second encryption is known. Although this notion may seem overly strong, natural cryptosystems (e.g., ElGamal, hash-proof-system-based cryptosystems) do satisfy this property. Indeed, under ElGamal a group element q is encrypted as $(g^r, g^{r \cdot sk} \cdot q)$, allowing the (encoded) randomness g^r be reused under a new secret key. Let $\mathcal{E} = (Gen, E, D, R)$ be a reproducible PKE scheme. Define $\mathcal{E}' = (Gen', E', D')$ as follows: $(pk', sk') \leftarrow Gen'$, where $sk' = r$ and $pk' = c = E_{pk}(0; r)$ (i.e., the secret key is a randomness string r and the public key is a dummy ciphertext formed under r); $E'_c(b)$ samples $(pk_1, sk_1) \leftarrow Gen$, computes $c' = R(c, b, sk_1)$ and returns (pk_1, c') (i.e., E'_c encrypts b by reusing the randomness underlying c); and $D'_r(pk_1, c')$ returns the bit b that $E_{pk_1}(b; r) = c'$. Intuitively, CPA security of $\mathcal{E}'$ follows from reproducibility and CPA security of $\mathcal{E}$. Moreover, the construction swaps the key and randomness spaces of $\mathcal{E}$, and so the task of securely encrypting randomness in $\mathcal{E}'$ reduces to that of securely self-encrypting the secret key in $\mathcal{E}$; this latter is the problem of *circular security*, a special case of the well-studied problem of *key-dependent-message* security [1–3,8,11–13,23]. The discussion above suggests a general technique for de-randomizing reproducible bit-encryption schemes, sketched below, which is the basis for all our subsequent constructions.

For $\mathcal{E} = (Gen, E, D, R)$ define $\mathcal{F} = C(\mathcal{E}) = (G, F, F^{-1})$ as follows. The domain space of F is the set of all pairs of public/secret keys generated under $Gen(1^n)$.

- G: To produce index/trapdoor keys (ik, tk), let $(pk, sk) \leftarrow Gen(1^n)$, set $ik = (pk, E_{pk}(0; r_1), \ldots, E_{pk}(0; r_l))$, for random r_i's, and set $tk = (r_1, \ldots, r_l)$.
- $F(\cdot, \cdot)$: On key $ik = (pk, c_1, \ldots, c_l)$ and domain input (pk', sk'), return $(pk', c'_1, \ldots, c'_l)$, where $c'_i = R(c_i, sk'_i, sk')$. (Here, sk'_i denotes the ith bit of sk'.)
- $F^{-1}(\cdot, \cdot)$: given trapdoor key $tk = (r_1, \ldots, r_l)$ and image point $(pk', c'_1, \ldots, c'_l)$, return $(pk', b_1 \ldots b_l)$, where b_i is the bit which satisfies $c'_i = E_{pk'}(b_i; r_i)$.

Correctness of $\mathcal{F}$ follows by the reproduction property of R. Also, since R is deterministic, so is the evaluation algorithm F. Finally, we take advantage of the fact that $\mathcal{E}$ is bit encryption to ensure efficient inversion for $\mathcal{F}$.

To discuss one-wayness we need the following definitions. For (pk, sk) output by Gen we refer to $E_{pk}(sk)$ as an *sk-self-encryption*. We call $\mathcal{E}$ *k-rec circularly secure* if no adversary can recover (with a nonnegligible chance) a random

sk from k independent sk-self-encryptions, and call $\mathcal{E}$ *k-ind circularly secure* if no adversary can distinguish between k independent sk-self-encryptions and encryptions of, say, zero. The notion of circular security in the literature is that of k-ind circular security, for unbounded k. For the construction above we show the following *tight* reduction.

Theorem 1. *If $\mathcal{E}$ is reproducible and 1-rec circularly secure then $C(\mathcal{E})$ is one-way.*

The reduction above is "security preserving" in the following sense: assuming $\mathcal{E}$ is reproducible, then $\mathcal{E}$ is 1-rec circularly secure iff $C(\mathcal{E})$ is one-way. Indeed, as we show next, by strengthening the condition of 1-rec circular security we achieve stronger forms of one-wayness.

A family of TDFs is called *k-wise one-way* [28] if one-wayness holds even if the given input is evaluated under k independently chosen functions.[1] More formally, $\mathcal{F} = (G, F, F^{-1})$ is called k-wise one-way, if $\mathcal{F}$'s *k-wise product*, defined as $F_{ik_1,\ldots,ik_k}(x) = (F_{ik_1}(x), \ldots, F_{ik_k}(x))$ is one-way. Rosen and Segev [28] showed the utility of this notion by giving a blackbox construction of CCA2-secure encryption based on k-wise one-way TDFs, for a sufficiently large k, simplifying a prior construction [26] based on lossy TDFs (LTDFs). Despite their utility, k-wise one-way TDFs (even for $k = 2$) are very strong primitives, whose only generic constructions so far have been based on LTDFs. Indeed, as shown by Vahlis [30], even 2-wise one-way TDFs cannot be constructed in a blackbox way from trapdoor permutations (TDPs).

Our TDF construction provides an easy means for obtaining k-wise one-way TDFs: we can generalize Theorem 1 to show the following

If $\mathcal{E}$ is reproducible and k-rec circularly secure then $C(\mathcal{E})$ is k-wise one-way.

To put our construction of k-wise one-way TDFs in context, we compare it to the LTDF-based construction [28]: the security reduction of [28] involves both statistical and computational arguments, allowing one to obtain only k-wise one-way TDFs for a priori fixed but arbitrarily large values of k (which does suffice for CCA2 encryption) from sufficiently lossy TDFs. Our reduction argument, on the other hand, is entirely computational, allowing us to obtain unbounded k-wise one-way TDFs (i.e., a TDF that is k-wise one-way for any value of k) from the full circular security assumption.

As for the base assumptions, the relationships among the circular-security notions we described is not well-understood (beyond the trivial ones). Under certain assumptions these notions become equivalent. For example, any *re-randomizable* 1-rec circularly-secure scheme is poly-ind circularly secure: this follows by considering that a 1-rec circularly-secure scheme is already poly-rec circularly secure (because of re-randomizability), and that any poly-rec circularly-secure scheme is also poly-ind circularly secure [29, Theorem 8]. For the rest of the introduction, however, for simplicity, we describe the results wrt full circular security.

[1] Actually, [28] chose another name for this particular notion, but we refer to it as k-wise one-wayness for simplicity.

We extend Construction C for the case in which the base scheme is t-circularly secure (i.e., circularly-secure wrt t keys): the input of each TDF is t pairs of public/secret keys, the index key contains $l \cdot t$ dummy ciphertexts, and the evaluation algorithm on $(pk_0, sk_0, \ldots, pk_{t-1}, sk_{t-1})$ returns $(pk_0, \ldots, pk_{t-1})$ along with $t \cdot l$ ciphertexts formed by encrypting each bit of sk_i under $pk_{(i+1 \mod t)}$ (deterministically) by reusing the randomness of the corresponding ciphertext of the index key.

Extracting Hardcore Bits. Given the TDFs built above, we may apply the general Goldreich-Levin (GL) theorem [20] to extract a hardcore bit. We would like to, however, avoid the use of the GL theorem for several reasons. First, the GL reduction, due to its generality, is not tight, while we would like to achieve CCA security with tight reductions. Second, for our deterministic encryption results we need to be able to extract many hardcore bits. Finally, since our base assumptions are strictly BB-stronger (by Vahlis's result) than one-way TDFs, we should look for more specialized methods. We sketch below two deterministic methods for extracting many hardcore bits with tight security reductions for our constructed TDFs. The first method applies to t-circular security and allows us to extract $\log((t-1)!)$ bits, with the advantage that it only increases the domain size. The second method allows us to extract any, a priori fixed, number of bits, but it enlarges other spaces as well.

First Method: A Cycle Hides its Ordering. For simplicity, we describe the idea for 3-circular security, showing how to extract a single hardcore bit. The idea is 3-circularly security implies no adversary can distinguish between the sequence $(E_{pk_1}(sk_2), E_{pk_2}(sk_3), E_{pk_3}(sk_1))$ and $(E_{pk_1}(sk_3), E_{pk_2}(sk_1), E_{pk_3}(sk_2))$. Now we augment our TDF construction described above (for t-circular security) so that the evaluation algorithm, besides $(pk_1, sk_1), (pk_2, sk_2), (pk_3, sk_3)$, also receives an additional bit b, used to dictate the ordering used to form the cycle. The inversion algorithm can open the ciphertexts, as before, and recover the bit b, by checking, say, whether the key encrypted under pk_1 is a secret key for pk_2 or for pk_3.[2] This technique extends to the t-circular security case for any $t > 3$, allowing us to "hide" a random ordering, providing $\log((t-1)!)$ hardcore bits.

Second Method. We describe the idea for 1-circular security. We extend construction C above to be parameterized over an integer $m = m(n)$ and to result in a TDF whose input now consists of triples (pk, sk, x), where $x \in \{0,1\}^m$. Moreover, we augment the index key to contain m added ciphertexts and let the trapdoor key contain their underlying randomness strings. Now $F(ik, (pk, sk, x))$ proceeds as before, but it also "encrypts" x in the process by again reusing randomness. For this TDF, we show that x remain pseudorandom even knowing $F(ik, (pk, sk, x))$. Finally, assuming the property that public keys under the base scheme are computed deterministically from their secret keys (plus perhaps some public parameters), we show how to obtain TDFs that hide a $(1 - o(1))$ fraction of their input bits.

[2] This, however, imposes a negligible inversion error.

CCA-secure Encryption. Using results on k-wise one-way TDFs with many hardcore bits,[3] we may now use the BB construction of Rosen and Segev [28] to build a many-bit CCA2-secure PKE from a reproducible, circularly secure bit-encryption scheme. Specifically, [28] gives a BB construction of CCA2-secure encryption from k-wise one-way TDFs, for $k \in \Omega(n)$; they also show that $k \in \omega(\log n)$ suffices for CCA1 encryption. Our CCA constructions, by relying on that of [28], result in schemes whose decryption functions query the encryption function of the base scheme. Gertner et al. [18] refer to such constructions as *non-shielding*, and show that there exist no *shielding* BB construction of CCA-secure from CPA-secure encryption. Since our base assumptions are BB-stronger than CPA security, it is natural to ask whether the non-shielding nature of our CCA2 construction is just an artifact of the construction of [28] or whether it is inherent. We were not able to answer this question for our encryption primitive, mainly because of the presence of the reproduction function. However, we are able to answer this wrt a weaker primitive than ours, which is a special case of *randomness-dependent-message-secure (RDMS)* encryption [7], which allows multiple bitwise-encryptions of a randomness string r under r itself as randomness (Formalized in Definition 2). Calling this new primitive RDMS encryption, we show that RDMS encryption is implied by our base assumptions, and also that it enables a non-shielding construction of CCA-secure encryption. We prove this by directly instantiating k-wise one-way TDFs under RDMS encryption. Next we observe that the shielding-BB impossibility result of [18] extends if the base scheme is an RDMS encryption primitive (Theorem 5). Indeed, it seems that this latter statement is true for most encryption primitives whose security requirements are defined wrt passive indistinguishability (i.e., no decryption oracles); see Sect. 4 for more details. Thus, we obtain an encryption primitive, wrt which a non-shielding BB CCA-secure construction is possible, but under which a shielding CCA-secure construction is BB impossible.

Deterministic Encryption (DE). Following [9], a deterministic l-bit-encryption scheme is called (λ, l)-IND secure if encryptions of any two (efficient) λ-sources (i.e., distributions with min-entropy λ) result in computationally indistinguishable ciphertexts. We formulate two extended notions of circular security, called (λ, l)-*entropy circular security* and *strong-*(λ, l)-*entropy circular security*, both of which require circular security hold even if the secret key $sk \in \{0,1\}^l$ is sampled from a λ-source distribution, while the strong-entropy version requires one more assumption, related to the public-key distribution.[4]

We show our TDF construction immediately gives us a (λ, l)-IND-secure DE scheme if the base scheme satisfies strong (λ, l)-entropy circular security. We also show, by appropriately choosing the parameters, the schemes of [11,12] provide strong-entropy circular security, meaning our generic transformation applies to these two schemes to obtain secure DE schemes, which explains the striking similarities between (especially) the DDH-based DE scheme of [9] and the scheme

[3] We note that our hardcore-security results hold not only for $\mathcal{F} = C(\mathcal{E})$, but also for $\mathcal{F}$'s k-wise products (under the respective assumptions). See Sect. 3.

[4] The notion of weak-entropy circular security was also considered by [13] in the context of KDM amplification.

of Boneh et al. [11]. We also note that the extra condition of strong-entropy circular security may be satisfied if, informally, the key-generation algorithm acts as a *strong extractor*, producing the public key from the secret key, taken as the source, based on a public parameter, taken as the seed. Similar structural assumptions are made in other settings, e.g., [32], to obtain DE schemes.

For weak-entropy circular security we also show how to obtain a secure DE scheme but with looser parameters, i.e., the (λ, l)-parameters of the base scheme are not maintained. We follow the so-called *encrypt-with-hardcore* technique, implicitly used in [4,6,9], and formalized in [17]. A high-level description of the idea is as follows. Assume $\mathcal{F} = (G, F, F^{-1})$ is a TDF with an associated hardcore function h producing $\Omega(n)$ hardcore bits, and we want to make $\mathcal{F}$ a secure DE scheme. Suppose we have the bonus that h preserves hardcore security even if x is sampled from a biased, high-min entropy distribution. Now we can build a DE scheme by encrypting the output of F using the hardcore bitstring under a randomized-encryption scheme $\mathcal{E}'$: namely, $E_{ik,pk}(x) = E'_{pk}(F(ik, x), h(x))$; decryption can be done using ik's trapdoor key and pk's secret key. Security of E comes from the fact that $(F(ik, x), h(x))$ is computationally indistinguishable from $(F(ik, x), r)$, so $h(x)$ is as good as a fresh randomness string. The only remaining issue is that E may require a longer randomness string, which, however, can be handled by applying a pseudorandom generator to $h(x)$.

Further Discussion. Since LTDFs [26] are the only generic assumption (to the best of our knowledge) that imply k-wise one-way TDFs, it is natural to ask about the relationship between LTDFs and our base primitive. We believe these notions are incomparable. First, under our encryption primitive, we are able to obtain a TDF that is k-wise one-way for unbounded k's; LTDFs are known to achieve bounded k-wise one-way TDFs, but this does not seem to generalize to the unbounded case, mainly due to the nature of LTDF-based proof techniques that also rely on statistical arguments. On the other hand, LTDFs have powerful statistical properties (i.e., losing information in lossy mode) which do not seem to be realizable under our assumptions. Choi and Wee [14], by abstracting the DDH-based TDF construction of [26], show how to obtain LTDFs from reproducible encryption that is homomorphic wrt both messages and randomness. For similar reasons our assumptions seem incomparable to those of [26].

We note that almost all BB CCA2-constructions, based on encryption or TDFs, are non-shielding [24,26,28], except for a few cases which rely on very powerful primitives, e.g., [10]. Intuitively, the non-shielding property of those constructions is used to do consistency checks on ciphertexts. It would be interesting to explore if there exist weaker encryption primitives (than those we consider) for which the BB separation of [18] is the best possible.

Our results show an alternative way (to those presented in [16,26]) of constructing DDH-based TDFs. Right now, by instantiating our TDF construction under the DDH-based circularly-secure scheme [11], we obtain no improvement in efficiency over existing constructions. This motivates the search for more efficient DDH-based circularly-secure schemes.

Finally, we discuss adaptations of Construction $C(\mathcal{E})$ to the case in which the secret-key space of $\mathcal{E}$ is a subset of the plaintext space $\mathcal{M}$ (which allows the secret

key to be encrypted as a whole) and reproducibility holds wrt $\mathcal{M}$. For this case we may substantially improve efficiency by having each index key contain only one ciphertext, whose randomness will be reused to self-encrypt the secret key (as a whole) given as input to the evaluation algorithm. To perform inversion, however, we would need to rely on one more assumption: it is efficiently possible to recover m from $E_{pk}(m;r)$ and r, for all pk, m and r. This last property by itself is satisfied by natural cryptosystems, e.g., ElGamal. Moreover, there is a standard way to make any CPA-secure scheme (for which $\{0,1\}^l \subseteq \mathcal{M}$) "one-shot" circularly secure; this transformation, however, does not (necessarily) maintain this last property. Thus, our results suggest the CPA-to-one-shot-circular transformation may be non-trivial (and interesting) if it is to maintain the last property.

2 Basic Notation and Definitions

Remark. Since we gave outlines of the proofs of most theorems in the introduction we defer the full proofs to the full version of the paper.

Notation. For a finite set S we use $x \leftarrow S$ to denote sampling x uniformly at random from S and denote by U_S the uniform distribution on S. If D is a distribution then $x \leftarrow D$ denotes choosing x according to D. We use the word PPT in this paper in the standard sense. We use $A(\ldots;r)$ to denote the deterministic output of PPT function A when the randomness is fixed to r, and use $x \leftarrow A(a_1, a_2, \ldots)$ to denote the distribution formed by outputting $A(a_1, a_2, \ldots; r)$ for a uniformly-random r. If $A(x_1, \ldots, x_m; r)$ outputs a tuple of strings, we let $A_i(x_1, \ldots, x_m)$ be the distribution formed by outputting the ith component of $A(x_1, \ldots, x_m)$. We denote the support set of a distribution D by $Sup(D)$, and write $x \in D$ to indicate $x \in Sup(D)$. We call $f : \mathbb{N} \to \mathbb{R}$ negligible if $f(n) < 1/P(n)$, for any polynomial P and sufficiently large n. We write $negl$ to denote unspecified negligible functions. We denote by f^{-1} the inverse of an injective function f. For two ensembles $X = \{X_i\}_{i\in\mathbb{N}}$ and $\{Y_i\}_{i\in\mathbb{N}}$ of random variables we say X is computationally indistinguishable from Y, denoted $X \equiv^c Y$, if for any bit-valued, PPT function D, we have $|\Pr[D(X_n) = 1] - \Pr[D(Y_n) = 1]| = negl(n)$. We write $X \equiv Y$ to mean X and Y are identically distributed. All functions, adversaries, distributions, etc., that appear in this paper, if not otherwise stated, are assumed to be efficiently computable/samplable. For $x, y \in \{0,1\}^*$ we use $|x|$ to denote the bit length of x, use x_i, for $1 \leq i \leq |x|$, to denote the ith bit of x, and use $x||y$ to denote the concatenation of x and y.

Trapdoor Functions. We first start by giving the standard definitions related to trapdoor functions and hardcore bits.

A collection, $\mathcal{F} = (G, F)$, of functions is defined as follows. The algorithm $G(1^n)$ returns a function index s, and the deterministic algorithm $F(s, \cdot)$ computes a function $f_s : D_n \to R_n$. We stress both the domain and range of f_s only depend on the security parameters, 1^n. We call $\{D_n\}$ the domain space of $\mathcal{F}$.

Assuming that $\mathcal{D} = \{\mathcal{D}_n\}$ is a distribution over $\{D_n\}$ and $h : D_n \to \{0,1\}^{p(n)}$ is a deterministic function, we define the following notions. We say $\mathcal{F}$ is $\mathcal{D}$-*one-way* if for any adversary $\mathcal{A}$, $\Pr[f_s(\mathcal{A}(s, f_s(x))) = f_s(x)] = negl(n)$, where the probability is computed over $s \leftarrow G(1^n)$, $x \leftarrow \mathcal{D}_n$ and $\mathcal{A}$'s coins.

We say that h is a $\mathcal{D}$-*hardcore function* for $\mathcal{F}$ if for any adversary $\mathcal{A}$,

$$\left|\Pr[\mathcal{A}(s, f_s(x), h(x)) = 1] - \Pr\left[\mathcal{A}(s, f_s(x), U_{\{0,1\}^{p(n)}}) = 1\right]\right| = negl(n),$$

where $s \leftarrow G(1^n)$ and $x \leftarrow \mathcal{D}_n$. We may omit $\mathcal{D}$, from $\mathcal{D}$-hardcore, etc., when it is clear from context. Next, we define TDFs and their k-wise products [28].

A *collection of injective trapdoor functions* (TDFs)[5] is given by three algorithms $\mathcal{F} = (G, F, F^{-1})$, where $G(1^n)$ randomly produces a pair (ik, tk) of *index/trapdoor keys*, the deterministic algorithm $F(ik, \cdot)$ computes an injective function $f_{ik} : D_n \to R_n$, and $F^{-1}(tk, \cdot)$ computes $f_{ik}^{-1}()$. We stress that the input domain of f_{ik} only depends on the security parameter 1^n. We may sometimes relax the definition by allowing a negligible inversion error. The k-wise product of $\mathcal{F}$, denoted $\mathcal{F}^{(k)} = \left(G^{(k)}, F^{(k)}\right)$, is defined as follows. The algorithm $G^{(k)}(1^n)$ runs $G(1^n)$ independently k times to output k index keys, $(ik_1, \ldots, ik_k)$; on input x, $F^{(k)}((ik_1, \ldots, ik_k), \cdot)$ returns $(F(ik_1, x), \ldots, F(ik_k, x))$.

Assume $\mathcal{F}$ is a TDF with domain $D = \{D_n\}$ and $\mathcal{D} = \{\mathcal{D}_n\}$ is a distribution on D. We say $\mathcal{F}$ is *k-wise* $\mathcal{D}$-*one-way* if $\mathcal{F}^{(k)}$ is $\mathcal{D}$-one-way.

Bit Encryption Schemes. All encryption schemes that appear throughout, if not explicitly stated, are *bit-encryption* schemes. In our applications we need to work with a more general notion of encryption schemes involving *public parameters*. A bit-encryption scheme $\mathcal{E} = (Param, Gen, E, Dec)$ is defined as follows. $Param$ on input 1^n outputs a random parameter, par. The *key-generation algorithm*, Gen on inputs 1^n and par generates a public/secret key $(pk, sk) \leftarrow Gen(1^n, par)$; we assume pk includes par, so we do not include par as input to other algorithms. The *encryption algorithm*, E, on inputs 1^n, pk, bit b and randomness $r \in \mathcal{R}_n$, outputs ciphertext $c = E_{pk}(b; r)$. The *decryption algorithm*, Dec, takes a secret key sk and ciphertext c, and deterministically outputs a bit $b = Dec_{sk}(c)$. For correctness, we require $\Pr[Dec_{sk}(E_{pk}(b)) = b] = 1$, for $par \leftarrow Param(1^n)$, $(pk, sk) \leftarrow Gen(1^n, par)$ and $b \leftarrow \{0,1\}$. We assume the following: for any fixed par, all secret keys output by $Gen(1^n)$ are bitstrings of the same length, and, whenever we are generating many public keys, all keys are generated wrt a single initial par. Thus, we make $Param$ implicit henceforth.

We say $\mathcal{E} = (Gen, E, Dec)$ is *CPA secure* if $(pk, E_{pk}(0)) \equiv^c (pk, E_{pk}(1))$, where pk is chosen according to $Gen(1^n)$. For $m \in \{0,1\}^*$, we extend E to define $E_{pk}(m) = (E_{pk}(m_1), \ldots, E_{pk}(m_{|m|}))$. If $\mathbf{r} = (r_1, \ldots, r_t)$ and $m \in \{0,1\}^t$ we write $E_{pk}(m; \mathbf{r}) = (E_{pk}(m_1; r_1), \ldots, E_{pk}(m_t; r_t))$.

We now give definitions for circular security. We say $\mathcal{E} = (Gen, E, Dec)$ is *k-rec t-circularly secure* if $\Pr[\mathcal{A}(pk_1, \ldots, pk_t, \mathbf{c}_1, \ldots, \mathbf{c}_k) = sk_1] = negl(n)$ for every adversary $\mathcal{A}$, where $(pk_1, sk_1), \ldots, (pk_t, sk_t) \leftarrow Gen(1^n)$ and for every $1 \le i \le k$

$$\mathbf{c}_i \leftarrow (E_{pk_2}(sk_1), E_{pk_3}(sk_2), \ldots, E_{pk_1}(sk_t));$$

[5] We use TDF to refer to a collection of injective trapdoor functions henceforth.

We say $\mathcal{E}$ is *k-ind t-circularly secure* if $\mathcal{E}$ is CPA secure and also it holds that $(\mathbf{c}_1, \ldots, \mathbf{c}_k) \equiv^c (\mathbf{c}'_1, \ldots, \mathbf{c}'_k)$, where

$$\mathbf{c}'_i \leftarrow \left(E_{pk_2}(0^l), E_{pk_3}(0^l), \ldots, E_{pk_1}(0^l)\right),$$

for every $1 \leq i \leq k$, and $l = |sk_1|$. Note that we add CPA security as a separate condition because otherwise the definition may be satisfied trivially, e.g., consider the encryption scheme under which the secret key is always the all-zero string and the encryption function is the identity function.

Henceforth, when we say k-rec circular security (or k-ind circular security) we are referring to the definition wrt a single pair of public/secret keys.

Definition 1. *We call* $\mathcal{E} = (Gen, E, Dec)$ reproducible *if there exists a deterministic function* R*, called the* reproduction function*, s.t. for any* (pk_1, sk_1)*,* $(pk_2, sk_2) \in Gen(1^n)$*,* $r \in \mathcal{R}_n$ *and* $b_1, b_2 \in \{0,1\}$*,*

$$R\left(pk_1, E_{pk_1}(b_1; r), b_2, pk_2, sk_2\right) = E_{pk_2}(b_2; r).$$

For simplicity we omit the inclusion of pk_1 *and* pk_2 *as inputs to* R*.*

3 Constructing TDFs and Hardcore Bits

TDFs From Reproducible Encryption. We begin by giving a construction that takes as input a reproducible bit-encryption scheme and produces a TDF. We then show how to achieve increasingly stronger guarantees of one-wayness for the constructed TDF from corresponding assumptions on the base encryption primitive. We present the construction adapted to the t-circular security case (i.e., circular security wrt t keys), meaning that we will obtain guarantees of one-wayness for the constructed TDF from t-circular security assumptions.

We use D^t to denote the t'th Cartesian power of a set D. If $\mathcal{D}$ is a distribution, $\mathcal{D}^t$ denotes the t-tuple formed by sampling t times independently from $\mathcal{D}$.

Construction 1. *Construction* C_1 *takes as input a reproducible bit-encryption scheme* $\mathcal{E} = (Gen, E, Dec, R)$ *and* $t = t(n)$ *and it outputs a TDF,* $\mathcal{F} = (G, F, F^{-1})$*, with domain space* D^t*, where* $D = Sup(Gen(1^n))$*. Let* $l = l(n)$ *be the length of a secret keys output by* $Gen(1^n)$*.*

- $G(1^n)$*: Let* $(pk, sk) \leftarrow Gen(1^n)$*, and form* $tk = (r_{1,1}, \ldots, r_{1,l}, \ldots, r_{t,1}, \ldots, r_{t,l})$*, for independent* $r_{i,j}$*'s, and* $ik = (pk, c_{1,1}, \ldots, c_{1,l}, \ldots, c_{t,1}, \ldots, c_{t,l})$*, where for* $1 \leq i \leq t$ *and* $1 \leq j \leq l$*,* $c_{i,j} = E_{pk}(0; r_{i,j})$*. Return* (ik, tk)
- $F\left((pk, c_{1,1}, \ldots, c_{t,l}), (pk_1, sk_1, \ldots, pk_t, sk_t)\right)$ *returns* $(pk_1, \ldots, pk_t, c'_{1,1}, \ldots, c'_{t,l})$*, where for* $1 \leq i \leq t-1$ *and* $1 \leq j \leq l$ *we set* $c'_{i,j} = R(c_{i,j}, b_{i,j}, sk_{i+1})$*, and* $c'_{t,j} = R(c_{t,j}, b_{t,j}, sk_1)$*, with* $b_{i,j}$ *being the* j*th bit of* sk_i*.*
- $F^{-1}((r_{1,1}, \ldots, r_{t,l}), (pk_1, \ldots, pk_t, c'_{1,1}, \ldots, c'_{t,l}))$*: Retrieve each* sk_i*, for* $1 \leq i \leq t$*, bit-by-bit by encrypting back both* 0 *and* 1 *with the provided randomness (and under the appropriate public key) and finding the matching bit.*

The TDF's completeness follows by reproducibility. We point out a few remarks. First, the efficiency of the search performed by the inversion algorithm relies on the fact that each ciphertext is hiding a single bit, encrypted under the randomness known to the inverter. Second, our construction is entirely blackbox, also accessing (during evaluation) the reproduction function. Third, our construction extends to the non-bit-encryption case, by still continuing to encrypt the secret key bit-by-bit, but by fixing a mapping from bits to two fixed plaintext messages; for this case, the one-wayness of the constructed TDF reduces to bit-wise circular security of the base scheme (wrt the fixed mapping).

Theorem 1. *Assume $\mathcal{E}$ is a reproducible bit-encryption scheme and $\mathcal{F}$ is the TDF built from $\mathcal{E}$ in Construction 1 based on integer t. Then, $\mathcal{E}$ is k-rec t-circularly secure if and only if $\mathcal{F}$ is k-wise $\mathcal{D}$-one-way, where $\mathcal{D} = (Gen(1^n))^t$. Moreover, the reductions are tight.*

Extracting Many Hardcore Bits. We present two deterministic methods for extracting many hardcore bits from the TDF presented in Construction 1, with tight and efficient reductions to the indistinguishability variants of circular security assumptions. The first method applies to t-circular security for $t \geq 2$, allowing us to directly extract $\log((t-1)!)$ bits, by expanding only the domain space by the same number of bits (but without affecting the sizes of the other system's parameters). The second method is less restrictive, allowing us to extract (from t-circular security, for any $t \geq 1$,) $m(n)$ hardcore bits, where m is an arbitrary but a priori fixed poly function, by increasing the domain space by $m(n)$ bits and the image, index-key and trapdoor-key spaces by poly factors of $m(n)$. In particular, by choosing the parameter m appropriately we obtain TDFs which hide a $1 - o(1)$ fraction of their input bits.

First Hardcore Extraction Method. We begin with some notation. Define $[t] = \{1, \ldots, t\}$. Let

$$S = \{f : [t] \to [t] \mid f \text{ is injective } \& \ \forall X, \text{ s.t. } \emptyset \subsetneq X \subsetneq [t], \{f(y) \mid y \in X\} \neq X\},$$

for which we have $|S| = (t-1)!$. Intuitively, each $f \in S$ defines a possible circular ordering of encrypting a sequence of t pairs of keys, by having pk_i encrypt $sk_{f(i)}$. The condition $\forall X \subsetneq [t], \{f(y) \mid y \in X\} \neq X$ guarantees that we have a single, full cycle. For example, it is not the case that pk_1 encrypts sk_2, pk_2 encrypts sk_1 and the remaining keys encrypt each other in a circular manner. Fix $\mathcal{O} : \mathbb{Z}_{(t-1)!} \to S$ to be an efficient index function defined using a canonical ordering of the elements of S. We will also write $\mathcal{O}(i, x)$ to denote $f_i(x)$, where f_i is the ith function according to the ordering. We also require that, for any $f \in S$, given $sq = \{(x, f(x)) \mid x \in [t]\}$, it is possible to efficiently compute the index of f according to the ordering[6], which we (by slightly abusing the notation) denote by $\mathcal{O}^{-1}(sq)$. We now proceed to describe the modified TDF construction and the associated hardcore function.

[6] Such an ordering for which we have such a function $\mathcal{O}$ can be defined by fixing an efficient way of enumeration.

Construction 2. *Let $\mathcal{E} = (Gen, E, Dec, R)$, t and D^t be as in Construction 1. The domain space of the TDF, $\mathcal{F} = (G, F, F^{-1})$, we build is now $(D^t, \mathbb{Z}_{(t-1)!})$.*

- $G(1^n)$*: As in Construction 1.*
- $F((pk, c_{1,1}, \ldots, c_{t,l}), (pk_1, sk_1, \ldots, pk_t, sk_t, u))$ *is computed as follows. Define* $(ind_1, \ldots, ind_t) = (\mathcal{O}(u,1), \ldots, \mathcal{O}(u,t))$*. Informally, the output will be* $pk_1, \ldots, pk_t$ *together with a cycle of encrypted keys, where* pk_i *encrypts the bits of* sk_{ind_i}*. Return* $(pk_1, \ldots, pk_t, c'_{1,1}, \ldots, c'_{t,l})$*, where, for* $1 \leq i \leq t$ *and* $1 \leq j \leq l$*,* $c'_{i,j} = R(c_{i,j}, b_{i,j}, sk_i)$*, with* $b_{i,j}$ *being the* j*th bit of* sk_{ind_i}*.*
- $F^{-1}((r_{1,1}, \ldots, r_{t,l}), (pk_1, \ldots, pk_t, c'_{1,1}, \ldots, c'_{t,l}))$*: do the following steps:*
  - *for each* $1 \leq i \leq t$*, recover the bitstring,* x_i*, encrypted under* pk_i *bit-by-bit as follows: to retrieve the* j*th bit of* x_i*, encrypt both* 0 *and* 1 *under* pk_i *using randomness* $r_{i,j}$ *and check the result against* $c'_{i,j}$*;*
  - *for each* $1 \leq i \leq t$*, let* ind_i*, where* $1 \leq ind_i \leq t$*, be the index for which it holds that* pk_{ind_i} *is the matching public key of* x_i*,*[7] *and let* $sk_{ind_i} = x_i$*. Form* $sq = \{(1, ind_1), \ldots, (t, ind_t)\}$*; return* $(pk_1, sk_1, \ldots, pk_t, sk_t, \mathcal{O}^{-1}(sq))$*.*

Hardcore Function: *For $\mathcal{F}$ given above we define $h\colon (D^t, \mathbb{Z}_{(t-1)!}) \to \mathbb{Z}_{(t-1)!}$ as $h(pk_1, sk_1, \ldots, pk_t, sk_t, u) = u$.*

Correctness of the new TDF follows immediately. Note that Construction 1 is a special case of Construction 2, by forming the encrypted cycle wrt the fixed function f: $f(1) = t; f(2) = 1; \ldots, f(t) = t-1$. In contrast, Construction 2 forms the encrypted cycle according to a random f (provided as input to the TDF), where, as we show below, the random choice of f is what is computationally hidden by the output. We now have

Theorem 2. *Assuming $\mathcal{E} = (Gen, E, Dec, Rep)$ is k-ind t-circularly-secure, it holds that $\mathcal{F}$ is k-wise one-way and h is a hardcore function for $\mathcal{F}^k$.*

Second Hardcore Extraction Method. The second construction allows us to extract any (a priori fixed) number of pseudorandom bits, where these bits are the last input block of the TDF.

Construction 3. *Let $\mathcal{E} = (Gen, E, Dec, R)$, t and D^t be as in Construction 1, and let $m = m(n)$ be an integer. The domain space of the TDF we build is $(D^t, \{0,1\}^m)$. We define $\mathcal{F} = (G, F, F^{-1})$ as follows.*

- $G(1^n)$*: Let* $(pk, sk) \leftarrow Gen(1^n)$*, and form* $tk = (r_{1,1}, \ldots, r_{t,l}, r_1, \ldots, r_m)$*, where* $r_{i,j}$*'s and* r_h*'s are independent randomness values, and form* $ik = (pk, \mathbf{c})$*, where* $\mathbf{c}$ *consists of* $t \cdot l + m$ *encryptions of zero under* pk *using* $r_{i,j}$*'s and* r_h*'s as randomness. Return* (ik, tk)*.*
- *Define* $F((pk, c_{1,1}, \ldots, c_{t,l}, c_1, \ldots, c_m), (pk_1, sk_1, \ldots, pk_t, sk_t, x))$ *to be equal to* $(pk_1, \ldots, pk_t, c'_{1,1}, \ldots, c'_{t,l}, c'_1, \ldots, c'_m)$*, where* $c'_{i,j} = R(c_{i,j}, b_{i,j}, sk_{i+1})$ *for* $1 \leq i \leq t-1$*,* $c'_{t,j} = R(c_{t,j}, b_{t,j}, sk_1)$ *and* $c'_h = R(c_h, x_h, sk_1)$*, where* $1 \leq h \leq m$*,* $1 \leq j \leq l$ *and* $b_{w,j}$ *is the* j*th bit of* sk_w*, for* $1 \leq w \leq t$*.*

[7] This can be done by encrypting many bits under the public key and decrypting them under a candidate secret key. This, however, results in a negligible inversion error.

– $F^{-1}((r_{1,1},\ldots,r_{t,l},r_1,\ldots,r_m),(pk_1,\ldots,pk_t,c'_{1,1},\ldots,c'_{t,l},c'_1,\ldots,c'_m))$: *as in the previous constructions.*

Hardcore Function: *For $\mathcal{F}$ given above, we let $h\colon (D^t,\{0,1\}^m)\to\{0,1\}^m$ be defined as $h(pk_1,sk_1,\ldots,pk_t,sk_t,x)=x$.*

Correctness of inversion is again evident, and we have security as follows.

Theorem 3. *Assuming $\mathcal{E}=(Gen,E,D,Rep)$ is k-ind t-circularly-secure, it holds that $\mathcal{F}$ is k-wise one-way and h is a hardcore function for $\mathcal{F}^k$.*

Remark 1. In many concrete settings, for a PKE $(Param,Gen,E,Dec)$, we have $Gen(1^n,par)\equiv(Pub_{par}(sk),sk)$, for a deterministic function Pub (recall par is output by $Param$): namely, the public key is obtained deterministically from the secret key and public parameters. We may now easily modify Construction 3, so that the index key also includes par and that the evaluation function no longer takes pk as input (so its entire input is a bitstring), by computing $pk=Pub_{par}(sk)$ by itself. Now letting $m\in\omega(t\cdot l)$ we obtain a TDF (from the assumptions stated in Theorem 3) hiding a $(1-o(1))$-fraction of its input bits.

4 Construction of CCA Secure Encryption

Rosen and Segev [28, Theorem 1] give a BB construction of CCA1-secure encryption from any $\omega(\log n)$-wise TDF and a BB CCA2-secure encryption from any $\Omega(n)$-wise TDFs. We may use our results and those of [28] to build CCA-secure encryption. For concreteness, we give the CCA1 construction here, which simplifies that obtained by directly instantiating [28] under our base encryption primitive. The construction for the CCA2 case is obtained similarly.

We fix the following notation. For $\mathbf{c}=(c_1,\ldots,c_m)$, $\mathbf{b}=(b_1,\ldots,b_m)$ we extend the reproduction function R so that $R(\mathbf{c},\mathbf{b},sk)$ denotes the sequence $(R(c_1,b_1,sk),\ldots,R(c_m,b_m,sk))$. We give the CCA1 construction below.

Suppose $\mathcal{E}=(Gen,E,Dec,R)$ has randomness space $\mathcal{R}_n$ and secret-key space $\{0,1\}^l$. We build a many-bit scheme $\hat{\mathcal{E}}=(\hat{Gen},\hat{E},\hat{Dec})$ as follows.

– $\hat{Gen}(1^n)$ samples $\mathbf{r}_0^1,\mathbf{r}_1^1,\ldots,\mathbf{r}_0^t,\mathbf{r}_1^t\leftarrow\mathcal{R}_n^l$, $(pk,sk)\leftarrow Gen(1^n)$ and returns $(\mathbf{pk},\mathbf{sk})$, where $\mathbf{pk}=(pk,E_{pk}(0^l;\mathbf{r}_0^1),E_{pk}(0^l;\mathbf{r}_1^1),\ldots,E_{pk}(0^l;\mathbf{r}_0^t),E_{pk}(0^l;\mathbf{r}_1^t))$ and $\mathbf{sk}=(\mathbf{r}_0^1,\mathbf{r}_1^1,\ldots,\mathbf{r}_0^t,\mathbf{r}_1^t)$;
– $\hat{E}_{\mathbf{pk}}(m)$ parses $\mathbf{pk}=(pk,\mathbf{c}_0^1,\mathbf{c}_1^1,\ldots,\mathbf{c}_0^t,\mathbf{c}_1^t)$, samples $(pk',sk')\leftarrow Gen(1^n)$, $u\leftarrow\{0,1\}^t$ and returns $(u,pk',E_{pk'}(m),\mathbf{c'}^1_{u_1},\ldots,\mathbf{c'}^t_{u_t})$, where, for $1\le i\le t$, $\mathbf{c'}^i_{u_i}=R(\mathbf{c}^i_{u_i},sk',sk')$ (Note that each $\mathbf{c'}^i_{u_i}$ is a self-encryption of sk');
– $\hat{Dec}_{\mathbf{sk}}(u,pk',c,\mathbf{c'}^1_{u_1},\ldots,\mathbf{c'}^t_{u_t})$ parses $\mathbf{sk}=(\mathbf{r}_0^1,\mathbf{r}_1^1,\ldots,\mathbf{r}_0^t,\mathbf{r}_1^t)$, lets sk_i, for each $1\le i\le t$, be the plaintext obtained bit-by-bit by "opening" $\mathbf{c'}^i_{u_i}$ relative to public key pk' and randomness vector $\mathbf{r}^i_{u_i}$, checks whether $sk_1=\cdots=sk_t$ (if this check fails it returns $\perp$), and returns $Dec_{sk_1}(c)$. Here by opening we mean finding the corresponding bit that encrypts to the given ciphertext under the specified randomness and public key.

In words, $\hat{E}$ samples (pk', sk') and a string u, and returns u, an encryption of m under pk' as well as t self-encrypted versions of sk', where the ith version reuses the randomness embedded in $\mathbf{c}^i_{u_i}$. We have the following theorem.

Theorem 4. *If $t \in \omega(\log n)$ and $\mathcal{E}$ is a reproducible, t-ind circularly-secure bit-encryption scheme, then $\hat{\mathcal{E}}$, constructed above, is CCA1 secure.*

The construction above is *non-shielding* [18], since the constructed decryption function queries the base encryption function.[8] By [18], there are no BB shielding constructions of CCA1-secure encryption from CPA-secure encryption. Since our base assumptions are strictly stronger than CPA security (at least in a BB sense), a natural question is whether or not it is possible to give a shielding construction based on our assumptions. At this point, we do not know the answer to this question, but as we show below, there exists an encryption primitive implied by our assumptions, based on which a non-shielding CCA1-construction is possible, but from which no *fully-blackbox*[9] shielding CCA1-construction is possible. Our new encryption primitive is an extension of CPA-secure encryption, asking that security holds even when encrypting certain *randomness-dependent messages*.

Definition 2. *A bit-encryption scheme $\mathcal{E} = (Gen, E, Dec)$ with randomness space $\{0,1\}^\rho$ is q-randomness-dependent-message (RDM) secure if*

$$\{E_{pk_1^1}(r_1; r), \ldots, E_{pk_\rho^1}(r_\rho; r)\}, \ldots, \{E_{pk_1^q}(r_1; r), \ldots, E_{pk_\rho^q}(r_\rho; r)\}$$
$$\equiv^c \{E_{pk_1^1}(0; r), \ldots, E_{pk_\rho^1}(0; r)\}, \ldots, \{E_{pk_1^q}(0; r), \ldots, E_{pk_\rho^q}(0; r)\},$$

where $r \leftarrow \{0,1\}^\rho$ and all public keys are chosen at random according to Gen. For better readability, we made the inclusion of the public keys implicit.

In the definition above, since we are encrypting the randomness string bitwise, we should use independent public keys for each encryption. Otherwise, an adversary can easily distinguish between the two distributions. Our definition is basically an adaptation of those of [7] to the bit-encryption case. We show below that this primitive is implied by our assumptions.

Given $\mathcal{E} = (Gen, E, Dec, R)$, define $\mathcal{E}' = (Gen', E', Dec')$, whose randomness space is the key space of $\mathcal{E}$, as follows: $Gen'(1^n)$ samples $(pk, sk) \leftarrow Gen(1^n)$ and $r \leftarrow \mathcal{R}_n$ and returns $pk = E_{pk}(0; r)$ and $sk = r$. The encryption $E'_c(b; (pk', sk'))$ returns $(pk', R(c, b, sk'))$; and, finally, $Dec'_r(pk', c')$ returns the bit b for which $E_{pk'}(b; r) = c'$. Using ideas described in Sect. 3 we can show, for any poly q, if $\mathcal{E}$ is q-ind circularly secure, then $\mathcal{E}'$ is q-RDM secure.

Next, we show q-RDM-secure encryption easily implies q-wise one-way TDFs. Let $\mathcal{E}$'s randomness space be $\{0,1\}^\rho$, and define TDF $\mathcal{TF} = (G, F, F^{-1})$ as follows. G runs $Gen(1^n)$ ρ times and returns $ik = (pk_1, \ldots, pk_\rho)$ and $tk = (sk_1, \ldots, sk_\rho)$; let F's domain space be $\mathcal{R}_n$ and define $F_{pk_1,\ldots,pk_\rho}(r)$ to equal $(F_{pk_1}(r_1; r), \ldots, F_{pk_\rho}(r_\rho, r))$. The inversion algorithm F^{-1} is defined in an obvious way. Now it is not hard to show if $\mathcal{E}$ is q-RDM secure, $\mathcal{TF}$ is q-wise one-way. A summary of the discussion above is the following.

[8] Due to lack of space, we refer the reader directly to [18] for a formal definition of shielding constructions.

[9] We are using the notion of fully-blackbox reductions as defined in [27].

Corollary 1. *For any $q \in \omega(\log n)$ there exists a shielding BB construction of CCA1-secure encryption from q-RDM-secure bit-encryption.*

We now show the BB separation of [18], stating that there are no shielding BB constructions of CCA1-secure encryption from CPA-secure encryption, extends even if the base scheme is RDM-secure, for any poly-bounded q. Combined with the corollary above, this gives us an encryption primitive which permits a non-shielding BB CCA1-secure construction, but from which no shielding BB CCA1-secure construction is possible. Specifically, [18] introduces a tuple of oracles $\mathcal{O} = (\mathbf{g}, \mathbf{e}, \mathbf{d}, \mathbf{w}, \mathbf{u})$, where $\mathcal{O}_1 = (\mathbf{g}, \mathbf{e}, \mathbf{d})$ model an idealized encryption scheme (when the oracle is chosen at random), and $\mathcal{O}_2 = (\mathbf{d}, \mathbf{w})$ are two security-weakening components, defined based on $\mathcal{O}_1$. They show that (*) for any candidate oracle-construction $\mathcal{E} = (Gen^{\mathcal{O}_1}, Enc^{\mathcal{O}_1}, Dec^{\mathbf{g},\mathbf{d}})$ there exists an oracle-adversary $\mathcal{A}^{\mathcal{O}}$, which is unbounded in time but poly-bounded in the number of oracle calls, that breaks the CCA1 security of $\mathcal{E}$ *almost always* (i.e., except for measure-zero of oracles). Thus, to rule-out shielding fully-BB constructions, it suffices to show that (**) for *almost any* selection of $\mathcal{O}$ (i.e., measure-one of oracles), $(\mathbf{g}, \mathbf{e}, \mathbf{d})$ is CPA-secure against any oracle-adversary $\mathcal{A}^{\mathcal{O}}$ with constraints mentioned above.[10] Therefore, to rule out shielding BB constructions of CCA2 secure encryption from a new encryption primitive, it suffices to prove (**) with respect to the new primitive. This is what we do below wrt RDM secure encryption. We first give the formal description of the oracles as in [18].

Definition 3. *([18]) Define ψ, a distribution on oracles $(\mathbf{g}, \mathbf{e}, \mathbf{d}, \mathbf{w}, \mathbf{u})$, defined for each $n \in \mathbb{N}$, as follows.*

- $\mathbf{g}\colon \{0,1\}^n \mapsto \{0,1\}^{3n}$ *is a random one-to-one function. Function* $\mathbf{g}$ *is considered as a key generator, with sk being the secret key and* $pk = \mathbf{g}(sk)$ *as the public key.*
- $\mathbf{e}\colon \{0,1\}^{3n} \times \{0,1\} \times \{0,1\}^n \mapsto \{0,1\}^{3n}$ *is a random one-to-one function.*
- $\mathbf{d}\colon \{0,1\}^n \times \{0,1\}^{3n} \mapsto \{0,1,\perp\}$ *is the unique function specified based on* $(\mathbf{g}, \mathbf{e})$*, where* $\mathbf{d}(sk, c) = b$ *if there exists* $r \in \{0,1\}^n$ *such that* $\mathbf{e}(\mathbf{g}(sk), b, r) = c$*; otherwise,* $\mathbf{d}(sk, c) = \perp$*.*
- $\mathbf{w}\colon \{0,1\}^{3n} \to \{0,1\}^{3n \times n} \cup \{\perp\}$ *is a random function sampled as follows. For* $\mathbf{w}(pk)$*, if* $\mathbf{g}^{-1}(pk) = \emptyset$ *then* $\mathbf{w}(pk) = \perp$*; otherwise, sample* $r_1, \ldots, r_n \leftarrow \{0,1\}^n$ *and return* $(\mathbf{e}(pk, sk_1, r_1), \ldots, \mathbf{e}(pk, sk_n, r_n))$*, where* $sk = \mathbf{g}^{-1}(pk)$*.*
- $\mathbf{u}\colon \{0,1\}^{3n} \times \{0,1\}^{3n} \mapsto \{\top, \perp\}$ *is a deterministic function which returns* $\top$ *if there exists sk, b and r such that* $\mathbf{g}(sk) = pk$ *and* $\mathbf{e}(pk, b, r) = c$*, and returns* $\perp$*, otherwise.*

For consistency, we may sometimes write $\mathbf{e}(pk, b, r)$ *and* $\mathbf{d}(sk, c)$*, respectively, as* $\mathbf{e}_{pk}(b; r)$ *and* $\mathbf{d}_{sk}(c)$*.*

We give the following theorem, a CPA version of which was proved in [18].

[10] We abuse notation somewhat here. By scheme $(\mathbf{g}, \mathbf{e}, \mathbf{d})$ we mean the oracle-aided scheme $(G^{\mathbf{g}}, E^{\mathbf{e}}, D^{\mathbf{d}})$ which just copies its oracle, e.g., $Gen(s)$ simply returns $\mathbf{g}(s)$.

Theorem 5. *For any adversary $\mathcal{A}$ and poly-bounded q, there exists a negligible function $negl$ such that*

$$\Pr_{\mathcal{O}=(\mathbf{g},\mathbf{e},\mathbf{d},\mathbf{w},\mathbf{u})\leftarrow\psi}\left[\Pr\left[\mathcal{A}^{\mathcal{O}}(ds_b)=b\right]\leq\tfrac{1}{2}+negl(n)\right]\geq 1-\tfrac{1}{2^{n/2}}, \tag{1}$$

where the inner probability is over b, the randomness of $\mathcal{A}$ and $ds_b \leftarrow \mathcal{DS}_b$, for

$$\mathcal{DS}_0 \equiv \{\mathbf{e}_{pk_1^1}(r_1;r),\ldots,\mathbf{e}_{pk_n^1}(r_n;r)\},\ldots,\{\mathbf{e}_{pk_1^q}(r_1;r),\ldots,\mathbf{e}_{pk_n^q}(r_n;r)\} \tag{2}$$

$$\mathcal{DS}_1 \equiv \{\mathbf{e}_{pk_1^1}(0;r),\ldots,\mathbf{e}_{pk_n^1}(0;r)\},\ldots,\{\mathbf{e}_{pk_1^q}(0;r),\ldots,\mathbf{e}_{pk_n^q}(0;r)\}, \tag{3}$$

in which $r \leftarrow \{0,1\}^n$ and the tuples $(pk_1^1,\ldots,pk_n^1)\ \ldots\ (pk_1^q,\ldots,pk_n^q)$ are formed, for every $1 \leq i \leq n$ and $1 \leq j \leq q$, by sampling $sk_i^j \leftarrow \{0,1\}^n$ and setting $pk_i^j = \mathbf{g}(sk_i^j)$.

We point out a few comments. First, the choice of $1-\frac{1}{2^{n/2}}$ for the quantity above is not strict; we made that choice just to be consistent with that of [18]. It can in fact be, for any constant $c < 1$, as large as $1-\frac{1}{2^{n/c}}$ by choosing appropriately the negligible function used to bound the inner probability in Eq. 1. Using standard techniques (especially applying the Borel-Cantelli lemma) [21], the inequality above may then be used to conclude that for measure-one of oracles $\mathcal{O} = (\mathbf{g},\mathbf{e},\mathbf{d},\mathbf{w},\mathbf{u})$, the scheme $(\mathbf{g},\mathbf{e},\mathbf{d})$ is q-RDM secure against all oracle-adversaries $\mathcal{A}^{\mathcal{O}}$.

By Theorem 5 and the results of [18], as discussed above, we have

Corollary 2. *For any $q \in \omega(\log n)$ there exists a non-shielding blackbox construction of CCA1 encryption from q-RDM-secure encryption. Moreover, for any poly-bounded q, there exists no shielding blackbox construction of CCA1 encryption from q-RDM-secure encryption.*

We note that it seems that one can generalize Corollary 2 to rule out the existence of shielding BB CCA1 constructions from a large class of encryption primitives whose security is defined in terms of indistinguishability against passive attacks (i.e., no decryption oracles). In other words, the BB separation generalizes to any (base) security requirement that is realized by an ideal encryption scheme $(\mathbf{g},\mathbf{e},\mathbf{d})$ in the presence of $(\mathbf{w},\mathbf{u})$; for example, Corollary 2 still holds true if RDM security is replaced with circular security.

5 Deterministic Encryption (DE) and Instantiations

We start by reviewing some basic facts related to entropy. The *min-entropy* of a distribution $\mathcal{D}$ is defined as $H_\infty(\mathcal{D}) = \min_{d\in\mathcal{D}} \log(1/\Pr[\mathcal{D}=d])$. If $l = H_\infty(\mathcal{D})$ we call $\mathcal{D}$ an *l-source*. We also recall the notion of *average min entropy*, formalized by Dodis et al. [15], defined as $\tilde{H}_\infty(\mathcal{X}|\mathcal{Y}) = -\log\left(E_{y\leftarrow Y}(2^{-H_\infty(\mathcal{X}|\mathcal{Y}=y)})\right)$.

DE Schemes. Since a DE scheme is syntactically the same as a TDF, we denote a DE scheme as $\mathcal{DE} = (G,F,F^{-1})$. We make a few assumptions in this section. We assume the conditions stated in Remark 1 hold for any randomized

encryption (RE) scheme used in this section: $Gen_1(1^n) \equiv Pub_{par}(sk)$, where Pub is a deterministic function; we often drop par. We use $l = l(n)$ to denote the length of a secret key of a RE scheme, and also the message length of a DE scheme.

We start by defining an extended notion of circular security, requiring circular security hold even if the secret key is sampled from a non-full-entropy distribution. For technical reasons, we need to allow some information about the secret key to be leaked, assuming the average min entropy of the secret key conditioned on the leaked information is high. The following definition generalizes a similar definition of [13] to the average case. We note it is possible to prove our results wrt the weaker definition of [13], but the proofs become more complex.

Definition 4. *We say a bit-encryption scheme $\mathcal{E} = (Gen, E, Dec)$ is (λ, l)-entropy circularly secure if for any joint distribution $(\mathcal{SK}, \mathcal{X})$, with $\tilde{H}_\infty(\mathcal{SK}|\mathcal{X}) \geq \lambda$, we have $(pk, E_{pk}(sk), E_{pk}(1), x) \equiv^c (pk, E_{pk}(0^l), E_{pk}(0), x)$, where $(sk, x) \leftarrow (\mathcal{SK}, \mathcal{X})$ and $pk = Pub(sk)$.*

Next we define a strengthening of the notion of [13], which adds the requirement that the public key distributions formed under high-entropy secret keys be computationally indistinguishable. This may be guaranteed if, e.g., Pub is a *strong extractor* [25], as is the case with known circularly-secure schemes [11,12].

Definition 5. *We say a bit-encryption scheme $\mathcal{E} = (Gen, E, Dec)$ is strongly-(λ, l)-entropy circularly secure if (a) for any λ-source $\mathcal{SK}$,*

$$(pk, E_{pk}(sk), E_{pk}(1)) \equiv^c (pk, E_{pk}(0^l), E_{pk}(0)),$$

where $sk \leftarrow \mathcal{SK}$ and $pk = Pub(sk)$; and (b) for any λ-sources $\mathcal{SK}_1$ and $\mathcal{SK}_2$, it holds that $Pub(\mathcal{SK}_1) \equiv^c Pub(\mathcal{SK}_2)$.

We now define our DE security notion, which is essentially the single-message, indistinguishability-based notion of [9]. See [9] for definitional equivalences.

Definition 6. *We say $\mathcal{DE} = (G, F, F^{-1})$ is secure wrt indistinguishability of λ-source inputs (shortly, (λ, l)-IND secure) if for any λ-sources $\mathcal{M}_0$ and $\mathcal{M}_1$, it holds $(ik, F_{ik}(\mathcal{M}_0)) \equiv^c (ik, F_{ik}(\mathcal{M}_1))$ where $(ik, tk) \leftarrow G(1^n)$.*

Now we show that by starting from a reproducible encryption scheme which provides strong (λ, l)-entropy circular security, Construction 1 immediately gives us a (λ, l)-IND secure deterministic scheme—i.e., it preserves the parameters.

Theorem 6. *Let $\mathcal{E} = (Gen, E, Dec, R)$ be a reproducible bit-encryption scheme and $\mathcal{DE} = C_1(\mathcal{E}, 1)$ be the DE scheme built in Construction 1 based on $\mathcal{E}$ and $t = 1$.[11] If $\mathcal{E}$ is strongly-(λ, l)-entropy circularly secure $\mathcal{F}$ is (λ, l)-IND secure.*

Next we show the "weaker" entropy circular security assumption also gives rise to DE schemes, but with looser security bounds. Our construction employs the encrypt-with-hardcore (EWH) technique, described in the introduction. To this end, we assume that the ciphertext space of our (base) encryption scheme is also a bitstring space, since our construction (by employing the EWH technique) results in double encryption. We give the main theorem below.

[11] Here we are working with a modified version of Construction 1 stated in Remark 1.

Theorem 7. *Let $\mathcal{E} = (Gen, E, Dec, R)$ be a reproducible (λ, l)-entropy circularly secure encryption scheme, with randomness space $\mathcal{R}_n = \{0,1\}^{p_r}$. There exists an $(l + p_r + u, 2l + p_r - \lambda)$-IND-secure deterministic encryption scheme, where $u \in \omega(\log n)$ is an arbitrary function.*

An outline of the proof follows, using notation given in the theorem above. The first step is to show we can use reproducibility of $\mathcal{E}$ to encrypt any arbitrarily-long message using a p_r-long randomness string, by reusing randomness across different public keys. Next, consider the TDF given by Construction 3, based on $t = 1$ and $m = l + p_r - \lambda$, and define $hc(sk, x) = (h, h(x))$, where $h\colon \{0,1\}^m \mapsto \{0,1\}^{p_r}$ is chosen from a *family of universal hash functions*, and show hc is a hardcore function for the TDF. Now to be able to apply the EWH method, we need to show, for $\mathcal{DS}_1 \equiv (h, h(x), E_{pk}(sk; \mathbf{r}_1), E_{pk}(x; \mathbf{r}_2), E_{pk}(0^l; \mathbf{r}_1), E_{pk}(0^{|x|}; \mathbf{r}_1))$ and $\mathcal{DS}_2 \equiv (h, y, E_{pk}(sk; \mathbf{r}_1), E_{pk}(x; \mathbf{r}_2), E_{pk}(0^l; \mathbf{r}_1), E_{pk}(0^{|x|}; \mathbf{r}_1))$, that $\mathcal{DS}_1 \equiv^c \mathcal{DS}_2$, where $y \leftarrow \{0,1\}^{p_r}$, $(sk, x) \leftarrow (\mathcal{SK}, \mathcal{X})$, $pk = Pub(sk)$ and $H_\infty(\mathcal{SK}, \mathcal{X}) \geq l + p_r + u$. (Also, $\mathbf{r}_1$ and $\mathbf{r}_2$ are chosen independently.) Now since $\tilde{H}_\infty(\mathcal{SK}|\mathcal{X}) \geq \lambda + u$ (which follows from standard average min-entropy facts) we may appeal to (λ, l)-entropy circular security of $\mathcal{E}$ to replace $E_{pk}(sk; \mathbf{r}_1)$, in both $\mathcal{DS}_1$ and $\mathcal{DS}_1$, with an all-zero encryption; in the next step we do the same for $E_{pk}(x; \mathbf{r}_1)$ (i.e., getting rid of the occurrences of x as a plaintext); and finally, using the facts that $\tilde{H}_\infty(\mathcal{X}|\mathcal{SK}) \geq p_r + u$, h is an *average-case extractor* and $u \in \omega(\log n)$, we replace $h(x)$ with a random string.

Instantiations. In the remainder of this section we briefly and informally review the scheme of Boneh et al. [11] (BHHO) and show it is reproducible. We defer the proof for the scheme of [12] as well as the proofs of entropy circular security to the full version.

Letting $\mathcal{G}$ be a *group scheme*, generate $\mathbb{G} \leftarrow \mathcal{G}$, $\mathbf{g} \leftarrow \mathbb{G}^l$ and set $par = (\mathbb{G}, \mathbf{g})$ and $o = |\mathbb{G}|$. Define (Gen, E, Dec) as follows. $Gen(1^n)$: samples $sk \leftarrow \{0,1\}^l$ and sets $pk = sk \cdot \mathbf{g}$ (where $\cdot$ denotes the inner product); $E_{pk}(g_1; r)$: samples $r \leftarrow \mathbb{Z}_o$ and returns $(\mathbf{g}^r, pk^r \cdot g_1)$, where $\mathbf{g}^r$ denotes element-wise exponentiation; and $D_{sk}(\mathbf{g}', g')$: clear from the encryption algorithm.To show reproducibility, we need to show given $pk_1 = sk_1 \cdot \mathbf{g}$, $c_1 = (\mathbf{g}^r, {pk_1}^r \cdot g_1)$, sk_2 and g_2, we can compute $(\mathbf{g}^r, {pk_2}^r \cdot g_2)$, where $pk_2 = sk_2 \cdot \mathbf{g}$, which is clear from the group properties. As for (strong)-(λ, l)-entropy circular security, we note that for the schemes [11,12] the fraction l/λ can be set arbitrarily large.

6 Conclusions and Open Problems

We gave generic constructions of several cryptographic primitives based on a general technique for de-randomizing reproducible bit-encryption schemes. For all the primitives we built it is already known that a BB construction from CPA-secure encryption alone is either impossible, or very difficult to find. We mention two main open problems that arise from our work. First, it would be interesting to see to if the BB result of [19] already separates TDFs from circularly-secure

encryption; showing this would imply that our reliance on an additional property, i.e., reproducibility, is unavoidable. Second, we would like to see whether the LWE-based circularly-secure scheme of Appleabaum et al. [2] can be used to instantiate our base assumptions.

References

1. Applebaum, B.: Key-dependent message security: generic amplification and completeness. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 527–546. Springer, Heidelberg (2011)
2. Applebaum, B., Cash, D., Peikert, C., Sahai, A.: Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 595–618. Springer, Heidelberg (2009)
3. Barak, B., Haitner, I., Hofheinz, D., Ishai, Y.: Bounded key-dependent message security. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 423–444. Springer, Heidelberg (2010)
4. Bellare, M., Boldyreva, A., O'Neill, A.: Deterministic and efficiently searchable encryption. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 535–552. Springer, Heidelberg (2007)
5. Bellare, M., Boldyreva, A., Staddon, J.: Randomness re-use in multi-recipient encryption schemeas. In: Desmedt, Y.G. (ed.) PKC 2003. LNCS, vol. 2567, pp. 85–99. Springer, Heidelberg (2003)
6. Bellare, M., Fischlin, M., O'Neill, A., Ristenpart, T.: Deterministic encryption: definitional equivalences and constructions without random oracles. In: Wagner [31], pp. 360–378
7. Birrell, E., Chung, K.-M., Pass, R., Telang, S.: Randomness-dependent message security. In: Sahai, A. (ed.) TCC 2013. LNCS, vol. 7785, pp. 700–720. Springer, Heidelberg (2013)
8. Black, J., Rogaway, P., Shrimpton, T.: Encryption-scheme security in the presence of key-dependent messages. In: Nyberg, K., Heys, H. (eds.) SAC 2002. LNCS, vol. 2595, pp. 62–75. Springer, Heidelberg (2003)
9. Boldyreva, A., Fehr, S., O'Neill, A.: On notions of security for deterministic encryption, and efficient constructions without random oracles. In: Wagner [31], pp. 335–359
10. Boneh, D., Canetti, R., Halevi, S., Katz, J.: Chosen-ciphertext security from identity-based encryption. SIAM J. Comput. **36**(5), 1301–1328 (2007)
11. Boneh, D., Halevi, S., Hamburg, M., Ostrovsky, R.: Circular-secure encryption from decision Diffie-Hellman. In: Wagner [31], pp. 108–125
12. Brakerski, Z., Goldwasser, S.: Circular and leakage resilient public-key encryption under subgroup indistinguishability. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 1–20. Springer, Heidelberg (2010)
13. Brakerski, Z., Goldwasser, S., Kalai, Y.T.: Black-box circular-secure encryption beyond affine functions. In: Ishai, Y. (ed.) TCC 2011. LNCS, vol. 6597, pp. 201–218. Springer, Heidelberg (2011)
14. Choi, S.G., Wee, H.: Lossy trapdoor functions from homomorphic reproducible encryption. Inf. Process. Lett. **112**(20), 794–798 (2012)
15. Dodis, Y., Ostrovsky, R., Reyzin, L., Smith, A.: Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. SIAM J. Comput. **38**(1), 97–139 (2008)

16. Freeman, D.M., Goldreich, O., Kiltz, E., Rosen, A., Segev, G.: More constructions of lossy and correlation-secure trapdoor functions. J. Cryptology **26**(1), 39–74 (2013)
17. Fuller, B., O'Neill, A., Reyzin, L.: A unified approach to deterministic encryption: new constructions and a connection to computational entropy. In: Cramer, R. (ed.) TCC 2012. LNCS, vol. 7194, pp. 582–599. Springer, Heidelberg (2012)
18. Gertner, Y., Malkin, T., Myers, S.: Towards a separation of semantic and CCA security for public key encryption. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 434–455. Springer, Heidelberg (2007)
19. Gertner, Y., Malkin, T., Reingold, O.: On the impossibility of basing trapdoor functions on trapdoor predicates. In: 42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14–17 October 2001, Las Vegas, Nevada, USA, pp. 126–135. IEEE Computer Society (2001)
20. Goldreich, O., Levin, L.A.: A hard-core predicate for all one-way functions. In: Johnson [22], pp. 25–32
21. Impagliazzo, R., Rudich, S.: Limits on the provable consequences of one-way permutations. In: Johnson [22], pp. 44–61
22. Johnson, D.S. (ed.): Proceedings of the 21st Annual ACM Symposium on Theory of Computing. ACM, New York (1989)
23. Malkin, T., Teranishi, I., Yung, M.: Efficient circuit-size independent public key encryption with KDM security. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 507–526. Springer, Heidelberg (2011)
24. Myers, S., Shelat, A.: Bit encryption is complete. In: 50th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2009, 25–27 October 2009, Atlanta, Georgia, USA, pp. 607–616. IEEE Computer Society (2009)
25. Nisan, N., Zuckerman, D.: Randomness is linear in space. J. Comput. Syst. Sci. **52**(1), 43–52 (1996)
26. Peikert, C., Waters, B.: Lossy trapdoor functions and their applications. In: Dwork, C. (ed.) STOC, pp. 187–196. ACM (2008)
27. Reingold, O., Trevisan, L., Vadhan, S.P.: Notions of reducibility between cryptographic primitives. In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 1–20. Springer, Heidelberg (2004)
28. Rosen, A., Segev, G.: Chosen-ciphertext security via correlated products. SIAM J. Comput. **39**(7), 3058–3088 (2010)
29. Rothblum, R.D.: On the circular security of bit-encryption. In: Sahai, A. (ed.) TCC 2013. LNCS, vol. 7785, pp. 579–598. Springer, Heidelberg (2013)
30. Vahlis, Y.: Two is a crowd? A black-box separation of one-wayness and security under correlated inputs. In: Micciancio, D. (ed.) TCC 2010. LNCS, vol. 5978, pp. 165–182. Springer, Heidelberg (2010)
31. Wagner, D.: Advances in Cryptology - CRYPTO 2008. LNCS, vol. 5157. Springer, Heidelberg (2008)
32. Wee, H.: Dual projective hashing and its applications — lossy trapdoor functions and more. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 246–262. Springer, Heidelberg (2012)