Adaptive Trapdoor Functions and Chosen-Ciphertext Security

Eike Kiltz[1], Payman Mohassel[2], and Adam O'Neill[3]

[1] CWI, Amsterdam, Netherlands
kiltz@cwi.nl
[2] University of Calgary, AB, Canada
pmohasse@cpsc.ucalgary.ca
[3] Georgia Institute of Technology, Atlanta, GA, USA
amoneill@cc.gatech.edu

Abstract

We introduce the notion of *adaptive* trapdoor functions (ATDFs); roughly, ATDFs remain one-way even when the adversary is given access to an inversion oracle. Our main application is the black-box construction of chosen-ciphertext secure public-key encryption (CCA-secure PKE). Namely, we give a black-box construction of CCA-Secure PKE from ATDFs, as well as a construction of ATDFs from correlation-secure TDFs introduced by Rosen and Segev (TCC '09). Moreover, by an extension of a recent result of Vahlis (TCC '10), we show that ATDFs are strictly *weaker* than the latter (in a black-box sense). Thus, adaptivity appears to be the weakest condition on a TDF currently known to yield the first implication.

We also give a black-box construction of CCA-secure PKE from a natural extension of ATDFs we call *tag-based* ATDFs that, when applied to our constructions of the latter from either correlation-secure TDFs, or lossy TDFs introduced by Peikert and Waters (STOC '08), yield precisely the CCA-secure PKE schemes in these works. This helps to unify and clarify their schemes. Finally, we show how to realize tag-based ATDFs from an assumption on RSA inversion not known to yield correlation-secure TDFs.

1 Introduction

Historically, the notion of one-way trapdoor functions (OW-TDFs) has played a central role in the study of cryptographic protocols, in particular for semantically-secure public-key encryption (PKE); see e.g. [**?**,**?**,**?**]. However, it is well-known that semantic security alone is not sufficient in many applications; rather, encryption must be secure against *active* adversaries, say, who can inject packets into the network and observe decryptions or actions taken based on them. As a result, resistance to so-called *chosen-ciphertext attacks* (CCA) [**?**] has become the "gold standard" for security of PKE.

But, whereas there is a simple, black-box construction of semantically secure PKE from OW-TDFs [**?**], the same is not true of CCA-secure PKE. Instead, early constructions were based on generic non-interactive zero-knowledge proofs [**?**]. This calls into question the applicability of the TDF concept in the design of CCA-secure PKE. Indeed, the most successful approach for designing practical

CCA-secure PKE schemes so far has been based on specific number theoretic assumptions (e.g., [?,?]) and algebraic primitives such as hash proof systems [?] or algebraic set systems [?], which bypass TDFs. However, Peikert and Waters [?], and subsequently Rosen and Segev [?], recently introduced novel strengthenings to the notion of OW-TDFs and showed that these *do* imply simple, black-box constructions of CCA-secure PKE.

Still, we find an underlying "theory" of such stronger TDFs and their relation to CCA-secure PKE lacking. To this end we put forth a notion of *adaptive* trapdoor functions and study its relations to CCA-secure PKE. Surprisingly, we find that adaptivity, a seemingly fundamental notion in the context of chosen-ciphertext security, serves to weaken the assumptions on a TDF needed to imply black-box CCA-secure PKE, as well as to unify and clarify the schemes of [?,?]. Moreover, it leads to new ones, realized from assumptions not known to imply the notions of [?,?].

1.1 Our Contributions

Adaptive trapdoor functions. The central notion we introduce are adaptive trapdoor functions (ATDFs). Loosely speaking, ATDFs remain one-way even when the adversary is given access to an inversion oracle, which it may query on points other than its challenge. We also introduce a natural extension we call tag-based adaptive trapdoor functions (TB-ATDFs), which in addition to the normal input also take a tag. For TB-ATDFs, the adversary may query its oracle on any point, but on a tag other than the challenge one. These notions are quite simple and intuitive but to the best of our knowledge have not appeared before. (There have, however, been similar notions that we discuss later.)

CCA-Secure PKE from ATDFs. As our first result, we give black-box constructions of CCA-secure PKE from both ATDFs and TB-ATDFs. While constructing CCA-secure PKE from TB-ATDFs is straightforward, constructing the former from ATDFs turns out to be more subtle. We apply the classical construction of one-bit PKE using the hardcore bit of the ATDF [?], but it is important here that the ciphertext not contain the message xor'ed with the latter; rather the message is encrypted *as* the hardcore bit itself. By a recent result of Myers and Shelat [?], this construction implies a black-box many-bit scheme as well. On the other hand, hybrid encryption permits a much more efficient direct construction of such a scheme in the case that the ATDF is a permutation or has linearly many simultaneous hardcore bits.

Construction of ATDFs. In the random oracle model [?], the notions of ATDF and TDF are equivalent.[1] To construct ATDFs in the standard model, we examine the relation of ATDFs and TB-ATDFs to the recently-introduced notions of correlated-product TDFs (CP-TDFs) [?] and lossy TDFs (LTDFs) [?].

[1] For example, a TDF defined as $f(x) := (g(x), H(x))$ is adaptive one-way if TDF g is one-way and H is modeled as a random oracle.

Intuitively, CP-TDFs remain one-way even if the adversary sees many independent instances of the TDF evaluated on the same input, and LTDFs are TDFs whose description is indistinguishable from that of a function that loses information about its input (i.e., has a bounded range). Inspired by the constructions of CCA-secure PKE in [**?**,**?**] (which are based on earlier work by Dolev et al. [**?**]), we show simple, black-box constructions of both ATDFs and TB-ATDFs from CP-TDFs. Since as shown in [**?**], LTDFs imply the latter,[2] this also gives us ATDFs and TB-ATDFs from LTDFs. However, we show that ATDFs and TB-ATDFs allow much more efficient direct constructions using an all-but-one TDF (ABO-TDF) [**?**] as well.

Notably, when we apply our general construction of CCA-secure PKE to our constructions of TB-ATDFs from CP-TDFs and lossy+ABO-TDFs, what we obtain are precisely CCA-secure PKE schemes of [**?**] and [**?**], respectively. This means that these works were implicitly constructing TB-ATDFs, and that the latter "abstracts out" a particular aspect of their constructions not formalized before. This unifies and clarifies their schemes from a conceptual standpoint and also leads to optimized constructions.

A black-box separation. Very recently, Vahlis [**?**] showed that there is no black-box construction of CP-TDFs from OW-TDFs. We observe here that his result extends to rule out a black-box construction of the former from ATDFs as well, by using the same "breaking" oracle. (This does not immediately rule out a black-box construction of CP-TDFs from TB-ATDFs, but we also rule this out by giving a construction of TB-TDFs from exponentially-hard ATDFs; the latter is separated from CP-TDFs by our extension of Vahlis's result as well.) Combined with the above-mentioned constructions, this means that, surprisingly, ATDFs and TB-ATDFs are *strictly weaker* than CP-TDFs and LTDFs. The relations between the different primitives are pictured in Figure 1. The figure also contains some related existing notions discussed below.

TB-ATDF from II-RSA. Finally, we show that TB-ATDFs are realizable from specific assumptions not known to imply CP-TDFs. Namely, we consider the "instance-independent" RSA assumption (II-RSA) introduced (in a more general form) by Paillier and Villar [**?**]. Roughly, our assumption says that solving an RSA challenge $y = x^e \bmod N$ remains hard even when the adversary is given access to an inversion oracle that on input (y', e') returns $y'^{1/e'} \bmod N$, where $e \neq e'$ are primes. We show that II-RSA gives rise to a TB-ATDF. This also leads to a very efficient CCA-secure RSA-based PKE scheme in the standard model (though based on an interactive, non-standard assumption).

[2] The original construction of [**?**] assumes "sufficient" lossiness; this result was recently refined by Mol and Yilek [**?**], who showed that losing a non-negligible fraction of a single bit suffices.

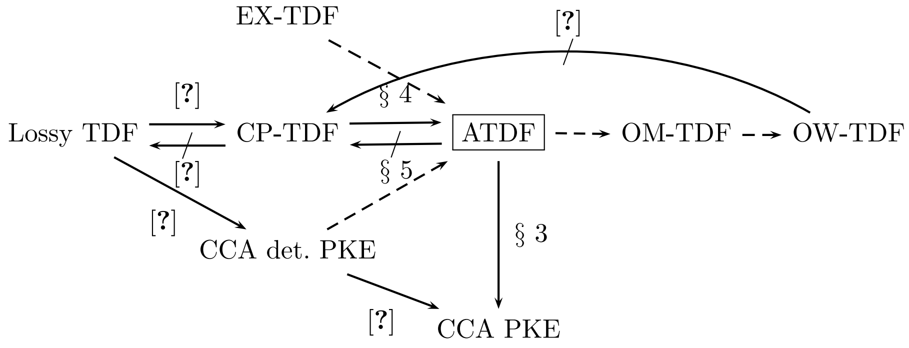

Fig. 1. Relations between the various security notions on trapdoor functions and CCA-secure PKE, centered around our new notion of adaptive trapdoor functions (ATDF). → is an implication while $\not\rightarrow$ is a black-box separation. Dashed lines indicate trivial implications mentioned in the introduction. The considered security notions for TDFs are: extractable TDF (EX-TDF), lossy TDF, correlated-product TDF (CP-TDF), one-more TDF (OM-TDF), one-way TDF (OW-TDF), and CCA-secure deterministic PKE.

1.2 Related Work

Related notions. Pandey et al. [?] introduced a notion they called "adaptive one-way functions," although their notion would be more accurately referred to as adaptive *tag-based* one-way functions. Besides the obvious difference of not having a trapdoor (the inversion oracle in their security experiment is unbounded), their notion differs from ours in that *it does not have a public key.* This is crucial for the applications of [?] to non-malleable commitment but also makes it much harder to construct. Indeed, they are not known to be realizable based on any standard assumptions.

Bellare et al. [?] made an earlier "adaptive assumption" on RSA, namely the One-More RSA assumption. A straightforward formalization of this security property to "one-more TDFs" (OM-TDFs)[3] yields a weaker primitive than ATDFs. In particular, it seems difficult based on the state-of-the-art to give a black-box construction of CCA-secure PKE (or ATDFs) from OM-TDFs. In [?], Canetti and Dakdouk define the notion of extractable trapdoor functions (EX-TDFs), which essentially says that no efficient adversary can compute $f(x)$ without "knowing" x (similar to the notion of plaintext-awareness for PKE [?]). This notion implies ATDFs but unfortunately no instantiation of EX-TDFs based on standard assumption is known (the authors only provide constructions of extractable one-way functions, without a trapdoor).

[3] Informally, a TDF is *one-more secure* if no efficient adversary can invert the TDF on $m+1$ challenges (obtained by querying a challenge oracle, for uniformly chosen preimages) given access to an inversion oracle that was queried up to m times.

In another line of work with very different motivation, Bellare et al. [?] introduced a strengthening to OW-TDF they called "deterministic encryption", which includes a CCA-secure variant. CCA-secure deterministic encryption (secure for encrypting a single message) can be viewed as a strengthening of ATDFs that additionally hides all partial information and allows for high-entropy input. CCA-secure deterministic encryption was constructed from CPA-secure PKE (satisfying a minor technical condition) in the random oracle model in [?] and in the standard model from LTDFs in [?]. We note that [?] gave a direct construction of CCA-secure PKE from CCA-secure deterministic encryption.

In the randomized encryption context, we mention the related notion of *tag-based* encryption [?,?,?]. Indeed, TB-ATDF can be viewed an analogue of selective-tag weakly CCA-secure PKE [?] in the TDF context. We also point out that the related notion of "one-way CCA" for encryption has surfaced before; see, e.g., [?]. (We stress that the difference is not just conceptual, as this notion is for *randomized* encryption.)

Work on black-box constructions. The importance of giving black-box constructions in cryptography is well-understood. A complementary line of work, starting with the seminal paper of Impagliazzo and Rudich [?], seeks to understand the limitations of such constructions. In the context of PKE, Choi et al. [?] recently showed a black-box construction of a *non-malleable* (i.e. NM-CPA) PKE scheme from any semantically-secure (i.e. IND-CPA) one, whereas [?] showed that there is no such construction of CCA-secure PKE whose decryption algorithm does not call the encryption algorithm of the starting scheme. In fact, CCA-secure PKE seems to be the remaining fundamental cryptographic task for which we know a non-black-box construction (from "enhanced" OW-TDPs) but not a corresponding black-box one. We hope that our work brings us closer to this goal.

1.3 Open Problems

Our works raises a number of interesting open problems. It may be interesting to consider other natural security notions for TDFs (e.g., non-malleability or q-bounded adaptivity [?]) and study their instantiation from standard assumptions, their implications for PKE, as well as their relation to existing notions from Figure 1. Furthermore, some of the relations in Figure 1, in particular between TDFs and ATDFs, are open.

Lossy TDFs are only known to be instantiable from *decisional* assumptions (such as DDH and QR), whereas we show that ATDFs are also instantiable from a computational assumption (though a non-standard and interactive one, namely II-RSA). An interesting open question is whether it is possible to instantiate ATDFs from more standard computational assumptions (such as RSA or CDH). One could also try to define a different security notion for TDFs, weaker than adaptivity, that admits instantiations from standard computational assumptions but still suffices for black-box CCA-secure PKE.

Finally, we are optimistic that ATDFs may be useful in the general context of black-box constructions of cryptograhpic primitives secure against adaptive attacks.

2 Preliminaries

Notation. If x and y are (binary) strings, then $x \| y$ denotes an encoding from which they are uniquely recoverable. If $k \in \mathbb{N}$ then 1^k denotes the string of k ones. If S is a set then $s \xleftarrow{\$} S$ denotes the operation of picking an element s of S uniformly at random. We write $\mathsf{A}(x, y, \ldots)$ to indicate that A is an algorithm (i.e., a Turing Machine) with inputs $x, y, \ldots$ and by $z \xleftarrow{\$} \mathsf{A}(x, y, \ldots)$ we denote the operation of running A with inputs $(x, y, \ldots)$ and letting z be the output. We write $\mathsf{A}^{\mathcal{O}_1, \mathcal{O}_2, \cdots}(x, y, \ldots)$ to indicate that A is an algorithm with inputs $x, y, \ldots$ and access to oracles $\mathcal{O}_1, \mathcal{O}_2, \ldots$. With PT we denote polynomial time and with PPT we denote probabilistic polynomial time.

CCA-secure PKE. A *public key encryption* scheme $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ with message space $\mathrm{MsgSp} = \mathrm{MsgSp}(k)$ consists of three PT algorithms, of which the first two, Kg and Enc, are probabilistic and the last one, Dec, is deterministic. Public/secret keys for security parameter $k \in \mathbb{N}$ are generated using $(pk, sk) \xleftarrow{\$} \mathsf{Kg}(1^k)$. Given such a key pair, a message $m \in \mathrm{MsgSp}$ is encrypted via $c \xleftarrow{\$} \mathsf{Enc}(pk, m)$; a ciphertext is decrypted by $m \leftarrow \mathsf{Dec}(sk, c)$. For correctness, we require that for all $k \in \mathbb{N}$, all messages $m \in \mathrm{MsgSp}$, it must hold that $\Pr[\mathsf{Dec}(sk, \mathsf{Enc}(pk, m)) = m] = 1$, where the probability is taken over the above randomized algorithms and $(pk, sk) \xleftarrow{\$} \mathsf{Kg}(1^k)$.

Let A be an adversary against PKE and define its $\mathsf{IND\text{-}CCA}$-*advantage* as

$$\mathbf{Adv}^{\mathrm{cca}}_{\mathsf{PKE},\mathsf{A}}(k) \;=\; 2 \cdot \Pr\left[b = b' : \begin{array}{c} (pk, sk) \xleftarrow{\$} \mathsf{Kg}(1^k) \\ (m_0, m_1, \mathrm{st}) \xleftarrow{\$} \mathsf{A}^{\mathcal{O}(sk,\cdot)}(pk) \\ b \xleftarrow{\$} \{0,1\} \;;\; c^* \xleftarrow{\$} \mathsf{Enc}(pk, m_b) \\ b' \xleftarrow{\$} \mathsf{A}^{\mathcal{O}(sk,\cdot)}(c^*, \mathrm{st}) \end{array} \right] - 1,$$

where $\mathcal{O}(sk, c) = \mathsf{Dec}(sk, c)$, and in the second phase ("guess phase") A is not allowed to query $\mathcal{O}(sk, \cdot)$ for the challenge ciphertext c^*. We also require that m_0 and m_1 are of the same length. (Here st is some arbitrary state information.) We say that PKE is $\mathsf{IND\text{-}CCA}$-secure if $\mathbf{Adv}^{\mathrm{cca}}_{\mathsf{PKE},\mathsf{A}}(\cdot)$ is negligible for all such PPT adversaries A.

3 Adaptive TDFs and CCA-Secure PKE Schemes

In this section, we introduce our notion of adaptive trapdoor functions (ATDFs) and an extension we call tag-based adaptive trapdoor functions (TB-ATDFs). We then show black-box constructions of CCA-secure PKE from these notions.

3.1 Adaptive Trapdoor Functions

Trapdoor Functions. Recall that a *trapdoor function* (TDF) is a triple of algorithms, where Tdg is probabilistic and on input 1^k generates an evaluation/trapdoor key-pair $(ek, td) \xleftarrow{\$} \mathsf{Tdg}(1^k)$, $\mathsf{F}(ek, \cdot)$ implements a function $f_{ek}(\cdot)$ over $\{0,1\}^k$ and $\mathsf{F}^{-1}(td, \cdot)$ implements its inverse $f_{ek}^{-1}(\cdot)$. Here we require TDFs to be injective. (Following [**?**], however, one can extend our results to poly-to-one TDFs as well.) Note that the above definition is purely functional and does not impose any security requirement.

One-wayness. First we recall the standard notion of one-wayness for trapdoor functions. Let A be an inverter and define its *OW-advantage* against TDF as

$$\mathbf{Adv}^{\mathrm{ow}}_{\mathsf{TDF},\mathsf{A}}(k) \;=\; \Pr\left[x = x' : \begin{array}{l} (ek, td) \xleftarrow{\$} \mathsf{Tdg}(1^k)\;;\; x \xleftarrow{\$} \{0,1\}^k \\ y \leftarrow \mathsf{F}(ek, x)\;;\; x' \xleftarrow{\$} \mathsf{A}(ek, y) \end{array}\right].$$

Trapdoor function TDF is *one-way* if $\mathbf{Adv}^{\mathrm{ow}}_{\mathsf{TDF},\mathsf{A}}(\cdot)$ is negligible for every PPT inverter A.

Adaptive one-wayness. Intuitively, adaptivity means that one-wayness holds even when the adversary may query an inverse oracle on points other than its challenge. Let $\mathsf{TDF} = (\mathsf{Tdg}, \mathsf{F}, \mathsf{F}^{-1})$ be a trapdoor function. Let A be an inverter and define its *AOW-advantage* against TDF as

$$\mathbf{Adv}^{\mathrm{aow}}_{\mathsf{TDF},\mathsf{A}}(k) \;=\; \Pr\left[x = x' : \begin{array}{l} (ek, td) \xleftarrow{\$} \mathsf{Tdg}(1^k)\;;\; x \xleftarrow{\$} \{0,1\}^k \\ y \leftarrow \mathsf{F}(ek, x)\;;\; x' \xleftarrow{\$} \mathsf{A}^{\mathsf{F}^{-1}(td,\cdot)}(ek, y) \end{array}\right],$$

where we demand that A does not query y to its oracle. Note that the behavior of oracle when queried on a y' outside the range of $\mathsf{F}(td, \cdot)$ is undefined; it returns whatever $\mathsf{F}^{-1}(td, y')$ does in this case (typically $\perp$). We say that TDF is *adaptive one-way* (or simply adaptive) if $\mathbf{Adv}^{\mathrm{atdf}}_{\mathsf{TDF},\mathsf{A}}(\cdot)$ is negligible for every such PPT inverter A.

Tag-based adaptive one-wayness. A *tag-based* TDF is a triple of algorithms $\mathsf{TDF}_{tag} = (\mathsf{Tdg}_{tag}, \mathsf{F}_{tag}, \mathsf{F}^{-1}_{tag})$ with associated *tag-space* $\mathit{TagSp}(k)$, where Tdg_{tag} is probabilistic and on input 1^k generates an evaluation/trapdoor key-pair $(ek, td) \xleftarrow{\$} \mathsf{Tdg}_{tag}(1^k)$. Furthermore, for every $t \in \mathit{TagSp}(k)$, $\mathsf{F}_{tag}(ek, t, \cdot)$ implements a function $f_{ek,t}(\cdot)$ over $\{0,1\}^k$ and $\mathsf{F}^{-1}_{tag}(td, t, \cdot)$ implements its inverse $f^{-1}_{ek,t}(\cdot)$. Let $\mathsf{A} = (\mathsf{A}_1, \mathsf{A}_2)$ be an inverter and define its *TB-AOW-advantage* against TDF_{tag} as

$$\mathbf{Adv}^{\mathrm{tb\text{-}aow}}_{\mathsf{TDF}_{tag},\mathsf{A}}(k) \;=\; \Pr\left[x = x' : \begin{array}{l} t \xleftarrow{\$} \mathsf{A}_1(1^k)\;;\; (ek, td) \xleftarrow{\$} \mathsf{Tdg}_{tag}(1^k) \\ y \leftarrow \mathsf{F}_{tag}(ek, t, x)\;;\; x' \xleftarrow{\$} \mathsf{A}_2^{\mathsf{F}^{-1}_{tag}(td,\cdot,\cdot)}(ek, t, y) \end{array}\right],$$

where we demand that A_2 does not make a query of the form $\mathsf{F}^{-1}_{tag}(td, t, \cdot)$ to its oracle. We say that TDF_{tag} is *tag-based adaptive one-way* if $\mathbf{Adv}^{\text{tb-atdf}}_{\mathsf{TDF}_{tag},\mathsf{A}}(\cdot)$ is negligible for every such PPT inverter A.

In the above experiment the "challenge tag" t is independent of ek and hence it may also be called selective-tag security (similar to selective-ID security for IBE schemes). Stronger variants of this security notion can be obtained by allowing the adversary choose the challenge-tag t adaptively.

We note that typically one requires the size of the tag-space to be super-polynomial. In fact, TB-ATDFs with polynomial-size tag-space can be constructed from any OW-TDF, but are not sufficient for our applications.

Relations between ATDFs and TB-ATDFs. Note that tag-based TDFs can be viewed as a specific type of TDF in which the first part of the input is output in the clear. Using this observation, it is not difficult to show that ATDFs and TB-ATDFs are equivalent under *exponential hardness*, meaning that if we start with an exponentially-hard version of one primitive it implies an exponentially-hard version of the other; see the full version for details. It is an open question whether ATDFs and TB-ATDFs are equivalent in general.

3.2 CCA-Secure PKE from ATDFs

Construction from ATDFs. We show how to construct a one-bit CCA-secure PKE scheme from an ATDF. By a recent result of Myers and Shelat [?], this implies a black-box construction of a many-bit scheme as well.

Let $\mathsf{TDF} = (\mathsf{Tdg}, \mathsf{F}, \mathsf{F}^{-1})$ be a TDF and $\mathsf{hc}(\cdot)$ be a hardcore bit, for example the Goldreich-Levin bit [?]. We construct PKE scheme $\mathsf{PKE}[\mathsf{TDF}] = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ with message-space $\{0,1\}$ as follows:

- **Key Generation**: On input 1^k, run $(ek, td) \xleftarrow{\$} \mathsf{Tdg}$ and return (ek, td).
- **Encryption**: On input ek, m, where $m \in \{0,1\}$ do:
  ```
  For $i = 1$ up to $k$:
      $x \xleftarrow{\$} \{0,1\}^k$ ; $h \leftarrow \mathsf{hc}(x)$ ; If $h = m$ then return $\mathsf{F}(ek, x)\|0$
  Return $m\|1$.
  ```
- **Decryption**: On inputs td and $c = c_1\|\mathsf{flag}$, if $\mathsf{flag} = 1$ then return c_1, else return $\mathsf{hc}(\mathsf{F}^{-1}(td, c_1))$.

It is clear that the above construction satisfies *correctness*. (Note that if the encryption algorithm happens to output the message in the clear it is still correctly decrypted, so this is a security, not a functionality, concern.) We now turn to security.

Theorem 1. *If* TDF *is adaptive one-way, then the* $\mathsf{PKE}[\mathsf{TDF}]$ *defined above is* $\mathsf{IND\text{-}CCA}$*-secure.*

The proof reduces $\mathsf{IND\text{-}CCA}$ security of the scheme to security of a hardcore bit by turning an adversary against the former into a distinguisher for the hardcore bit that is given k independent samples, and then applying a hybrid argument.

We note that as a consequence, security of the scheme is only loosely related to security of the underlying hardcore bit (losing a factor $1/k$).

Proof (of Theorem 1). Given an adversary A against the PKE scheme, we transform its IND-CCA experiment via a sequence of games:

- **Game** G_1: The IND-CCA experiment.
- **Game** G_2**:** Instead of computing the hardcore bits using $\mathsf{hc}(\cdot)$, the encryption algorithm encrypts the challenge message by picking a uniformly random bit on each iteration of the for-loop. That is, the second line in the for-loop is replaced with "$h \xleftarrow{\$} \{0,1\}$."
- **Game** G_3**:** If the for-loop in the encryption algorithm completes its execution (without satisfying the $h = m$ condition), instead of returning the challenge message in the clear, it simply returns $\perp$ to the adversary. That is, the last line in the encryption algorithm is replaced with "Return $\perp$."

For $i \in \{1, 2, 3\}$, let $\Pr[\mathsf{A}^{G_i} \Rightarrow b]$ denote the probability that A outputs the challenge bit b when executed in Game G_i (taken over the coins of the game and of A).

We first claim that if there is an inverter A against TDF such that $\Pr[\mathsf{A}^{G_1} \Rightarrow b] - \Pr[\mathsf{A}^{G_2} \Rightarrow b]$ is non-negligible, then so is $\mathbf{Adv}^{\mathrm{aow}}_{\mathsf{TDF},\mathsf{A}}$. To show this, it suffices by a standard hybrid argument and security of $\mathsf{hc}(\cdot)$ to give a k-sample distinguisher D against $\mathsf{hc}(\cdot)$ whose advantage is non-negligible in this case. That is, D is given an input $ek, (y_1, h_1), \ldots, (y_k, h_k)$ where $y_i = \mathsf{F}(ek, x_i)$ and either $h_i = \mathsf{hc}(x_i)$ or is a uniformly random bit for all $1 \le i \le k$; D also has oracle access to $\mathsf{F}^{-1}(td, \cdot)$, which it may query on any y such that $y \neq y_i$ for all $1 \le i \le k$. Define D on inputs $ek, (y_1, h_1), \ldots, (y_k, h_k)$ as follows:

- Run A on input ek. When A makes a decryption query $c = c_1 \| \mathsf{flag}$, if $\mathsf{flag} = 1$ then respond with c_1 and otherwise $\mathsf{hc}(\mathsf{F}^{-1}(c_1))$. Let (m_1, m_2, st) be the output of A.
- Choose $b \xleftarrow{\$} \{0,1\}$ and find the least i^* such that $h_{i^*} = m_b$. If no such i^* exists, then set $c^* \leftarrow m_b \| 1$. Otherwise, set $c^* \leftarrow y_{i^*} \| 0$.
- Run A on inputs (c^*, st). When A makes a decryption query $c = c_1 \| \mathsf{flag}$, if $\mathsf{flag} = 1$ respond with c_1 and otherwise $\mathsf{hc}(\mathsf{F}^{-1}(c_1))$. Let b' be the output of A. Return 1 if $b = b'$ and 0 otherwise.

To show that D has the claimed property, note that the only way it can fail to give a perfect simulation of either Game G_1 or G_2 is if A makes a query of the form $y_i \| 0$ for some $1 \le i \le k$. It suffices to bound the probability of this when A is executed as in Game G_1, as follows. In the case $c^* = y_{i^*} \| 0$, A does not query $y_{i^*} \| 0$ by definition, and the probability it queries $y_i \| 0$ for any $i \neq i^*$ is at most $(k-1)q/2^{k-1}$ (where q an upper-bound on its number of decryption queries), where we use the fact that conditioning on $\mathsf{hc}(x_i) = 1 - m_b$ reduces the min-entropy of each x_i by at most 1 bit and $\mathsf{F}(ek, \cdot)$ is an injection. A similar analysis pertains to the case $c^* = m_b \| 0$. Overall, the probability is at most $kq/2^{k-1}$.

We next claim that $\Pr[\mathsf{A}^{G_2} \Rightarrow 1] - \Pr[\mathsf{A}^{G_3} \Rightarrow 1] \le 2^{-k}$. This follows by using the fact that in Game G_2 the hardcore bits used to encrypt the challenge message have been replaced with uniformly random ones.

Finally, observe that $\Pr[\mathsf{A}^{G_3} \Rightarrow b] = 1/2$, since in this game A gets no information about b. Combining the above gives the theorem.

Construction from TB-ATDF. Our construction of CCA-secure PKE from a TB-ATDF is much simpler. It additionally makes use of a strongly one-time unforgeable signature scheme (see e.g. [?] for the definition). For simplicity, we give the construction below for the case of 1-bit messages. It is easy to extend it to a many-bit scheme, essentially by concatenating many applications of the TB-ATDF under independent inputs but the same tag.

Let $\mathsf{TDF}_{tag} = (\mathsf{Tdg}_{tag}, \mathsf{F}_{tag}, \mathsf{F}_{tag}^{-1})$ be a tag-based TDF and let $\mathsf{hc}(\cdot)$ be a hardcore bit. Let $\mathsf{OTS} = (\mathsf{K}, \mathsf{S}, \mathsf{V})$ be a signature scheme whose verification keys are contained in the tag-space of TDF_{tag}. We construct PKE scheme $\mathsf{PKE}[\mathsf{TDF}_{tag}, \mathsf{OTS}] = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ with message-space $\{0,1\}$ as follows:

- **Key Generation**: On input 1^k, run $(ek, td) \xleftarrow{\$} \mathsf{Tdg}_{tag}$ and return (ek, td).
- **Encryption**: On input ek, m where $m \in \{0,1\}$, run $(sk, vk) \xleftarrow{\$} \mathsf{K}(1^k)$ and choose $x \xleftarrow{\$} \{0,1\}^k$. Set $y_1 \leftarrow \mathsf{F}_{tag}(ek, vk, x)$ and $y_2 \leftarrow \mathsf{hc}(x) \oplus m$; also, set $\sigma \leftarrow \mathsf{S}(sk, y_1 \| y_2)$. Return $y_1 \| y_2 \| vk \| \sigma$.
- **Decryption**: On inputs td and $y = y_1 \| y_2 \| vk \| \sigma$, if $\mathsf{V}(vk, \sigma) = 1$ then set $x \leftarrow \mathsf{TDF}_{tag}(td, vk, y_1)$ and return $\mathsf{hc}(x) \oplus y_2$; , otherwise return $\perp$.

We have the following theorem.

Theorem 2. *If* TDF_{tag} *is adaptive one-way and* OTS *is one-time strongly unforgeable, then then* $\mathsf{PKE}[\mathsf{TDF}_{tag}, \mathsf{OTS}]$ *is* IND-CCA-*secure.*

The proof is straightforward and hence omitted.

Optimizations. Our construction of CCA-secure PKE from ATDFs can be simplified and made much more efficient if the given ATDF is a permutation or has linearly many simultaneous hardcore bits. Namely, in this case one can use the ATDF as a key-encapsulation mechanism (KEM) for an IND-CCA-secure symmetric encryption scheme.

Additionally, for some *specific* hardcore bits one may be able to sample uniformly from the set $\{x \in \{0,1\}^k \mid \mathsf{hc}(x) = b\}$ more efficiently than by repeated sampling of the uniform distribution on $\{0,1\}^k$. (Indeed, this is the case for the universally-hardcore Goldreich-Levin bit [?].) This translates to a corresponding efficiency improvement in the scheme.

Our construction of CCA-secure PKE from TB-ATDFs can also be made much more efficient if the given TB-ATDF is a permutation (for every tag) or has linearly many simultaneous hardcore bits. The idea is to first construct a selective-tag weakly CCA secure tag-PKE scheme in the sense of [?] by using the TB-ATDF as a KEM for a one-time CPA-secure symmetric encryption scheme. Then, as shown in [?], we can apply the transform of Boneh et al. [?] to obtain a CCA-secure PKE scheme, which uses only symmetric-key primitives.

4 Constructing ATDFs from Stronger TDFs

Inspired by the constructions of CCA-secure PKE in [?,?], we show that both ATDFs and TB-ATDFs can be constructed in a simple black-box manner from correlated-product TDFs [?]. As shown in [?], lossy TDFs (LTDFs) [?] imply CP-TDFs, thus by our result above they imply ATDFs and TB-ATDFs too. However, we are able to give much more efficient direct construction in combination with an all-but-one TDF (ABO-TDF) as defined by [?].

4.1 Constructions from Correlated-Product TDFs

One-wayness under correlated-product. We first recall the notion of one-wayness under correlated product [?]. Let $\mathsf{TDF} = (\mathsf{Tdg}, \mathsf{F}, \mathsf{F}^{-1})$ be a trapdoor function, and let $\mathcal{C}_t$ be such that $\mathcal{C}_t(1^k)$ is distributed over $\{0,1\}^{tk}$ for a polynomial $t = t(k)$. Let A be an inverter and define its *$\mathcal{C}_t$-CP-advantage* against TDF as

$$\mathbf{Adv}^{\mathrm{cpow}}_{\mathsf{TDF},\mathsf{A}}(k) \;=\; \Pr\left[\begin{array}{r} \\ (x_1,\ldots,x_t) \\ = (x'_1,\ldots,x'_t) \\ \end{array} : \begin{array}{l} (ek_i, td_i) \xleftarrow{\$} \mathsf{Tdg}(1^k), 1 \le i \le t\,; \\ (x_1,\ldots,x_t) \xleftarrow{\$} \mathcal{C}_t(1^k)\,; \\ y \leftarrow (f_{ek_1}(x_1),\ldots,f_{ek_t}(x_t))\,; \\ (x'_1,\ldots,x'_t) \xleftarrow{\$} \mathsf{A}(ek_1,\ldots,ek_t,y) \end{array}\right].$$

We say that TDF *one-way under $\mathcal{C}_t$-correlated-product* if $\mathbf{Adv}^{\mathrm{cpow}}_{\mathsf{TDF},\mathsf{A}}(\cdot)$ is negligible for any PPT inverter A.

The cannonical $\mathcal{C}_t$ considered by [?] is such that $x_1 = x_2 = \ldots = x_t$, where x_1 is random. We call TDFs secure in this sense *one-way under t-correlated-product* (t-CP-TDF).

Construction of ATDFs. Let $\mathsf{TDF}_1 = (\mathsf{Tdg}_1, \mathsf{F}_1, \mathsf{F}_1^{-1})$ be a TDF, where we assume wlog (by suitable padding) that $\mathsf{TDF}_1(ek,\cdot)$ has a fixed output length $n = n(k)$. We construct $\mathsf{TDF} = (\mathsf{Tdg}, \mathsf{F}, \mathsf{F}^{-1})$ as follows:

- **Key Generation**: On input 1^k, let $(ek_0, td_0) \xleftarrow{\$} \mathsf{Tdg}(1^k)$ and for all $b \in \{0,1\}$ and $1 \le i \le n$ set $(ek_i^b, td_i^b) \xleftarrow{\$} \mathsf{Tdg}_1(1^k)$. Let $ek \leftarrow (ek_0, (ek_1^0, ek_1^1), \ldots, (ek_n^0, ek_n^1))$ and $td \leftarrow (td_0, (td_1^0, td_1^1), \ldots, (td_n^0, td_n^1))$. Return (ek, td).
- **Evaluation**: On inputs ek, x, return $\mathsf{F}_1(ek_0, x) \| \mathsf{F}_1(ek_1^{b_1}, x) \| \ldots \| \mathsf{F}_1(ek_n^{b_n}, x)$, where b_i denotes the ith bit of $\mathsf{F}(ek_0, x)$ for $1 \le i \le n$.
- **Inversion**: On inputs td and $y = y_0 \| y_1 \| \ldots \| y_n$, let $x \leftarrow \mathsf{F}_1^{-1}(td_0, y_0)$. Return x if $x = \mathsf{F}^{-1}(td_i^{b_i}, y_i) = \mathsf{F}_1^{-1}(td_0, y_0)$ for $0 \le i \le n$, where b_i denotes the ith bit of y_0, otherwise return $\perp$.

We have the following theorem.

Theorem 3. *If* TDF_1 *is a* $(n+1)$*-CP-TDF then* TDF *is an ATDF.*

Proof. Given an adversary A against TDF, we describe below an adversary B against TDF_1 such that $\mathbf{Adv}^{(\mathrm{n+1})\text{-}\mathrm{cpow}}_{\mathsf{B},\mathsf{TDF}_1} = \mathbf{Adv}^{\mathrm{aow}}_{\mathsf{A},\mathsf{TDF}}$.

On inputs $ek_1, \ldots, ek_{n+1}, y$ where $y = (\mathsf{F}_1(ek_1, x_1), \ldots, \mathsf{F}_1(ek_{n+1}, x_{n+1}))$, B sets $ek_0 \leftarrow ek_1$ and $ek_i^{b_i} \leftarrow ek_{i+1}$ for all $1 \leq i \leq n$, where b_i denotes the ith bit of $\mathsf{F}_1(ek_1, x_1)$. It then chooses $(ek_i^{1-b_i}, td_i^{1-b_i}) \xleftarrow{\$} \mathsf{Tdg}_1(1^k)$ for all $1 \leq i \leq n$. It runs A on inputs ek, y for ek defined as in the key generation algorithm of TDF. When A makes an inversion query $y' = y'_0 \| y'_1 \| \ldots \| y'_n$, B chooses an index i such that $b'_i \neq b_i$, where b'_i denotes the ith bit of y'_0. (As we argue below, such i must exist.) It sets $x' \leftarrow \mathsf{F}^{-1}(td_i^{b'_i}, y'_i)$. If $\mathsf{F}(ek, x') = y'$ then it returns x' to A and otherwise returns $\perp$. Finally, when A halts B returns its output.

It is clear that B satisfies the desired property. To finish the proof it remains to argue that index i used in answering A's inversion queries always exists. But this follows directly from injectivity of $\mathsf{F}(ek_0, \cdot)$ and the fact that A is not allowed to make an inversion query equal to its challenge.

Remarks. We note that it is possible to make the scheme more efficient by additionally using a universal one-way family (aka. TCR) of hash functions [**?**]. Then, the "selector" bits $b_1, \ldots, b_n$ in the construction are replaced with the bits of the hash of $\mathsf{F}(ek_0, x)$. We also note that following [**?**] it is possible to give a construction based on a CP-TDF allowing a slightly weaker correlation among the inputs.

Construction of TB-ATDFs. The above construction of ATDFs can easily be modified to give a construction of TB-ATDFs as well. The difference is that in the "selector" bits $b_1, \ldots, b_n$ are replaced with the bits $t_1, \ldots, t_n$ of the tag t. Notably, when we apply our construction of CCA-secure PKE from TB-ATDFs given in Section 3) to the resulting TB-ATDF, we obtain precisely the CCA-secure PKE scheme of [**?**].

4.2 Constructions from Lossy and All-but-One TDFs

We first recall the notion of lossy TDFs and their generalization called all-but-one TDFs from [**?**].

Lossy TDFs. A (k, ℓ)-*LTDF* is a quadruple $\mathsf{LTDF} = (\mathsf{LTdg}, \mathsf{LTdg}', \mathsf{LF}, \mathsf{LF}^{-1})$ of algorithms, where the triple $(\mathsf{LTdg}, \mathsf{LF}, \mathsf{LF}^{-1})$ is a TDF on $\{0,1\}^k$. We require that (1) the function $\mathsf{LTDF}(ek', \cdot)$ has a range of size at most 2^ℓ (where $\ell = \ell(k)$) for every ek', and (2) the keys ek, ek' are computationally indistinguishable, over the choice of $(ek, td) \xleftarrow{\$} \mathsf{LTdg}(1^k)$ and $ek' \xleftarrow{\$} \mathsf{LTdg}'(1^k)$.

All-but-one TDFs. An (k, ℓ)-*ABO-TDF* with branch-space $\{0,1\}^{n=n(k)}$ is a triple $\mathsf{ABO} = (\mathsf{ABO\text{-}Tdg}, \mathsf{ABO\text{-}F}, \mathsf{ABO\text{-}F}^{-1})$ of algorithms, where for every $r \neq r' \in \{0,1\}^n$, the triple $(\mathsf{ABO\text{-}Tdg}(1^k, r), \mathsf{ABO\text{-}F}(r', \cdot, \cdot), \mathsf{ABO\text{-}F}^{-1}(r', \cdot, \cdot))$ is a TDF on $\{0,1\}^k$ (the "lossy branch" r is passed as an input to $\mathsf{ABO\text{-}Tdg}$). We further require (1) for every $r \in \{0,1\}^n$ and ek', the function $\mathsf{ABO\text{-}F}(r, ek', \cdot)$ has range-size at most 2^ℓ (where $\ell = \ell(k)$), and (3) for every $r \neq r' \in \{0,1\}^n$, the keys ek_1, ek_2 are computationally indistinguishable, over $(ek_1, td_1) \xleftarrow{\$} \mathsf{ABO\text{-}Tdg}(1^k, r)$ and $(ek_2, td_2) \xleftarrow{\$} \mathsf{ABO\text{-}Tdg}(1^k, r')$.

Construction of ATDFs. Our construction simplifies the construction of CCA-secure deterministic encryption given in [?]. Let $\mathsf{LTDF} = (\mathsf{LTdg}, \mathsf{LTdg}', \mathsf{LF}, \mathsf{LF}^{-1})$ be a (k, ℓ_1)-LTDF and let $\mathsf{ABO} = (\mathsf{ABO\text{-}Tdg}, \mathsf{ABO\text{-}F}, \mathsf{ABO\text{-}F}^{-1})$ be a (k, ℓ_2)-ABO-TDF; for simplicity we assume its branch-space is $\{0,1\}^n$ for $n = n(k)$. Let $T : R \to (\{0,1\}^n \setminus \{0^n\})$ be a hash function, where R denotes the range of $\mathsf{LF}(ek, \cdot)$. We construct $\mathsf{TDF}[\mathsf{LTDF}, \mathsf{ABO}, T]$ as follows.

- **Key Generation**: On input 1^k, run $(ek_{\text{ltf}}, td_{\text{ltf}}) \xleftarrow{\$} \mathsf{LTdg}(1^k)$ and $(ek_{\text{abo}}, td_{\text{abo}}) \xleftarrow{\$} \mathcal{F}_{\text{abo}}(1^k, 0^n)$. Return $((ek_{\text{ltf}}, ek_{\text{abo}}), (td_{\text{ltf}}, td_{\text{abo}}))$.
- **Evaluation**: On inputs $(ek_{\text{ltf}}, ek_{\text{abo}})$ and $x \in \{0,1\}^k$, set $y_1 \leftarrow \mathsf{LF}(ek_{\text{ltf}}, x)$ and $y_2 \leftarrow \mathsf{ABO\text{-}F}(T(y_1), ek_{\text{abo}}, x)$. Return $y_1 \| y_2$.
- **Inversion**: On inputs $(td_{\text{ltf}}, td_{\text{abo}})$ and $y = y_1 \| y_2$, set $x \leftarrow \mathsf{LF}^{-1}(td, y_1)$. If $y_2 = \mathsf{ABO\text{-}F}(T(y_1), ek_{\text{abo}}, x)$ then return x, otherwise return $\perp$.

We have the following theorem.

Theorem 4. *If $\ell_1 + \ell_2 = k - \omega(\log k)$ and T is TCR, then $\mathsf{TDF}[\mathsf{LTDF}, \mathsf{ABO}, T]$ defined above is an ATDF.*

The proof follows [?] and is given in the full version.

Construction of TB-ATDFs. Similarly to our construction of ATDF from CP-TDF, the above construction can be adapted to construct a tag-based ATDF instead. The difference is that in the evaluation algorithm, instead of evaluating the all-but-one TDF at branch $T(y_1)$, it is evaluated at branch t, where the latter is the input tag. As before, when we apply our general construction of CCA-secure PKE from TB-ADTFs given in Section 3 to the resulting TB-ATDF, we obtain precisely the CCA-secure PKE scheme of [?].

5 On the Complexity of Adaptive TDFs

In this section, we show that there is no black-box construction of CP-TDFs from either ATDFs or TB-ATDFs; combined with the results of Section 4, this shows that the latter are (surprisingly) strictly *weaker* primitives (in a black-box sense). We then show that TB-ATDFs can be realized based on an assumption on RSA inversion not known to imply CP-TDFs.

5.1 A Black-Box Separation

Very recently, Vahlis [?] showed that there is no black-box construction of CP-TDFs from one-way TDFs. We observe here that his proof extends to rule out a black-box construction of CP-TDFs from either ATDFs or TB-ATDFs.

Theorem 5. *There is no black-box construction of CP-TDFs from ATDFs or TB-ATDFs.*

The theorem actually follows by extending Vahlis's proof to rule out a black-box construction of CP-TDFs from *exponentially-hard* ATDFs. Since as discussed in Section 3, TB-ATDFs are implied by exponentially-hard TDFs, this rules out a black-box construction of CP-TDFs from TB-ATDFs as well.

Since Vahlis's proof is rather technical we avoid explaining its details here. Instead, we describe the high-level ideas and point out a minor change needed to give our claimed result.

Similar to most black-box separation results, in order to show that there is no black-box construction of primitive P_1 from primitive P_2, the proof starts by defining an *ideal oracle* O (the ideal version of P_2), and a *break oracle* B. One then shows that (i) there exist an adversary A that breaks any construction of P_1, with the help of a small number of queries to B and (ii) P_2 can be securely realized using the ideal oracle O, even when the adversary is given access to B.

Oracle O. The ideal oracle O is essentially an ideal trapdoor permutation (as described in several previous works [**?**]). Roughly speaking, O is defined as a triple of functions (g, e, d) sampled uniformly at random from the set of all functions with the following property: g maps trapdoors to public keys; $e(pub, \cdot)$ is an independent permutation for every public key pub, and $d(pri, \cdot)$ inverts $e(pub, \cdot)$ if pri is the trapdoor corresponding to pub. One may assume that trapdoors, public keys, and inputs are all of the same length, i.e. equal to the security parameter. Also note that there is no need to explicitly define d as it is determined given the definitions of g and e.

It is easy to see that oracle O is an ATDF; in fact, it is an *exponentially-hard* ATDF. However, as pointed out in [**?**], O is also correlation secure as the permutations for every public key is chosen independently and uniformly at random.

oracle B. Oracle B is specially designed to break the security of a CP-TDF. It takes as input a triple of circuits (G^O, E^O, D^O) which are candidates for a correlation secure TDF, two public keys PUB_1, PUB_2 and the values $E(PUB_1, x)$ and $E(PUB_2, x)$. The naive solution would be to let oracle B return x. However, this would make oracle B too powerful and would allow an adversary to break the security of any ideal TDF by letting the two public keys be $pub_1 = pub_2$. This problem is solved by requiring that the public keys of O encoded in PUB_1 are *distinct* from those encoded in PUB_2. An additional problem is caused by the fact that the adversary can make queries that contain invalid public keys, while detecting invalid keys by oracle B can render it too powerful. This issue is resolved by requiring the adversary to provide a partial oracle $O' = (g', e', d')$ that is defined on a small part of the domain of (g, e, d) such that relative to O', PUB_1 and PUB_2 are valid public keys.

We refer the reader to [**?**] for a more formal description of oracles O and B. The following claims (proven in [**?**]), complete the argument.

Claim 1. ([**?**]) There exist an adversary that only makes polynomially many queries (in the security parameter) to oracles O and B, and breaks the security of any CP-TDF function with non-negligible probability.

Claim 2. Let $\mathsf{TDF} = (G^O, E^O, D^O)$ be the trapdoor function that simply forwards its inputs to $O = (g, e, d)$. For any adversary A that makes polynomially many queries to oracles B and O, A's advantage in breaking TDF in the ATDF game is negligible.

In [**?**], Claim 2 is proven for the case when A is playing the OW-TDF game. However, the proof easily extends to the case of adaptive TDFs. Particularly, the bulk of the proof consists of describing a simulator S that simulates the answers for queries made to oracle B. For consistency purposes, S keeps a list O^* of all the query/answers made to the challenge function $e(pub^*, \cdot)$ where pub^* is the challenge public key. In case of ATDFs, S needs to do the same for any query $e^{-1}(pub^*, \cdot)$ made to the inversion oracle. The rest of the proof stays the same.

Note that, in the above discussion, we did not restrict the running time of the adversaries. Instead, we only required that the number of queries they make to the oracles is small. It is however easy to bring things to the world of polynomial-time adversaries by giving everyone access to a PSPACE oracle (see [**?**]).

5.2 Tag-based ATDF from an Assumption on RSA Inversion

To further demonstrate the usefulness of our new notions, we show that TB-ATDFs are realizable from an assumption on RSA inversion not known to imply a CP-TDF.

Instance-independent RSA [**?**,**?**]. The instance-independent RSA assumption (II-RSA) speaks to the difficulty of solving the RSA problem — that is, computing e-th roots modulo $N = pq$ — even if given access to an oracle that computes e'-th roots modulo N for $e' \neq e$. Of course, some additional restriction on the exponents is necessary for this assumption to hold; in what follows we require that e, e' are primes. To define the assumption formally, let the tuple of algorithms $(\mathsf{RSA_g}, \mathsf{RSA}, \mathsf{RSA}^{-1})$ be defined in the natural way with the exception that the exponent e is no longer generated by the key generation step. That is, on input 1^k, algorithm $\mathsf{RSA_g}$ generates (ek, td) where $ek = N = pq$, and $td = (p, q)$ for two uniformly chosen $k/2$-bit primes p, q. Moreover, we require p, q to be *safe* primes, meaning $(p-1)/2, (q-1)/2$ are also prime. On inputs $e \in \mathbb{Z}^*_{(p-1)(q-1)}$, $x \in \mathbb{Z}^*_N$ and N, algorithm RSA returns $c = x^e \mod N$. On inputs $(p, q), e, y$, algorithm RSA^{-1} computes $d \leftarrow e^{-1} \mod (p-1)(q-1)$ and returns $y^d \mod N$. Let $n = n(k)$ be an integer. For an inverter $\mathsf{A} = (\mathsf{A}_1, \mathsf{A}_2)$ define its *II-RSA advantage for n* as

$$\mathbf{Adv}^{\text{II-RSA}}_{\mathsf{A},n}(k) \;=\; \Pr\left[x = x' : \begin{array}{r} e \xleftarrow{\$} \mathcal{P}_n \;;\; (ek, td) \xleftarrow{\$} \mathsf{RSA_g}(1^k) \\ x \xleftarrow{\$} \mathbb{Z}_N \;;\; y \leftarrow \mathsf{RSA}(ek, e, x) \\ x' \xleftarrow{\$} \mathsf{A}_2^{\mathsf{RSA}^{-1}(td,\cdot,\cdot)}(ek, e, y) \end{array} \right]$$

where here and in what follows $\mathcal{P}_n$ denotes the set of all n-bit primes and we require that A_2 only makes queries of the form $\mathsf{RSA}^{-1}(td, e', y')$ for primes $e' \neq e$.

We say that the $\mathsf{II\text{-}RSA}$ *holds for n* if $\mathbf{Adv}^{\text{II-RSA}}_{\mathsf{A},n}(\cdot)$ is negligible for every such PPT inverter A.

Discussion. $\mathsf{II\text{-}RSA}$ was first conjectured (in a more general form) by Paillier and Villar [**?**], whose work was concerned with showing that several RSA-based schemes *cannot* be proven secure in the standard model. More recently, Chevallier-Mames and Joye [**?**] observed that $\mathsf{II\text{-}RSA}$ can be used to *construct* CCA-secure encryption as well. We note that [**?**] actually considered the assumption parameterized by a *fixed* "challenge" e (e.g., $e = 3$). We follow the formulation [**?**] and choose e at random from the set of all primes of a given length.

Prime Sequence Generator. Our construction uses the "prime sequence generator" of [**?**], which for any $n \in \mathbb{N}$ with $k \geq (n + 1)/2$ probabilistically constructs an efficiently computable, (with overwhelming probability) injective map $\mathsf{phash}_n \colon \{0, 1\}^k \to \mathcal{P}_n$. First, one chooses a random $2(n + 1)^2$-wise-independent function $Q \colon \{0, 1\}^k \times \{1, \ldots, 2(n+1)^2\} \to \{0, 1\}^n$ using the standard polynomial evaluation construct over $\mathbb{F}_{2^{\kappa+1}}$. Then for $t \in \{0, 1\}^k$, we define $\mathsf{phash}_n(t)$ to be the first prime in the sequence $Q(t, 1), \ldots, Q(t, 2(n + 1)^2)$.

Tag-based ATDF from II-RSA. Let phash_n be as defined above for $k \geq (n+1)/2$. We construct a tag-based ATDF $\mathsf{TDF}_{tag}[\mathsf{phash}_n] = (\mathsf{Tdg}_{tag}, \mathsf{F}_{tag}, \mathsf{F}^{-1}_{tag})$ with tag-space $\{0, 1\}^k$ as follows:

- **Key Generation**: On input 1^k, return $(ek, td) \xleftarrow{\$} \mathsf{RSA}_{\mathsf{g}}(1^k)$.
- **Evaluation**: On inputs x, $ek = N$, and tag $t \in \{0, 1\}^k$, return $\mathsf{RSA}(ek, \mathsf{phash}_n(t), x)$.
- **Inversion**: On inputs y, $td = (p, q)$ and tag $t \in \{0, 1\}^k$, return $\mathsf{RSA}^{-1}(td, \mathsf{phash}_n(t), y)$.

We have the following theorem.

Theorem 6. *Let phash_n be as above. If the $\mathsf{II\text{-}RSA}$ assumption holds for n then $\mathsf{TDF}_{tag}[\mathsf{phash}_n]$ defined above is a TB-ATDF (in fact, it is a TB-ATDP).*

We stress that the use of the "prime sequence generator" in the construction does not introduce any unproven assumption.

Proof. (Sketch.) Given an adversary A against $\mathsf{TDF}_{tag}[\mathsf{phash}_n]$, we consider two games, which we call G_1 and G_2. Game G_1 is just the TB-ATDF experiment with A against $\mathsf{TDF}_{tag}[\mathsf{phash}_n]$. For Game G_2, we modify the inversion oracle to return $\perp$ whenever A makes an inversion query on a tag t' such that $\mathsf{phash}_n(t') = \mathsf{phash}_n(t)$, where t is the challenge tag. For $i \in \{1, 2\}$, let $\Pr\left[\, \mathsf{A}^{G_i} \Rightarrow x \,\right]$ denote the probability that A returns the challenge input x when executed in G_i.

First, we claim that $\Pr\left[\, \mathsf{A}^{G_1} \Rightarrow x \,\right] - \Pr\left[\, \mathsf{A}^{G_2} \Rightarrow x \,\right] \leq 2^{-\Omega(k)}$. This follows from the analysis of the prime sequence generator in [**?**], who show that with probability at least $1 - 2^{-\Omega(n)}$ over the choice of Q in its construction, the set $\{\mathsf{phash}_n(t) \,:\, t \in \{0, 1\}^k\}$ contains 2^k *random* and *distinct* n-bit primes.

Next, we claim that we can construct an adversary B_2 against $\mathsf{II\text{-}RSA}$ such that $\mathbf{Adv}^{\text{II-RSA}}_{\mathsf{B}_2} = \Pr\left[\, \mathsf{A}^{G_2} \Rightarrow x \,\right]$, which completes the proof. Note that an adversary against $\mathsf{II\text{-}RSA}$ receives the challenge exponent e "from the outside," so we need a way of "programming" the prime sequence generator at a given point. For this we can use the ideas of [?], who show that for any $t^* \in \{0,1\}^n$ and random $e^* \in \mathcal{P}_n$, it is possible to construct the polynomial $Q = Q_{t^*,e^*}$ for the prime sequence generator in such a way that $\mathsf{phash}_n(t^*) = e^*$ and that for every t_0^*, t_1^*, the distribution of these Q's are $2^{-\Omega(n)}$-close.

An efficient CCA-secure RSA-based PKE scheme. The above construction of TB-ATDP leads to a very efficient CCA-secure RSA-based PKE scheme in the standard model. Namely, we apply the "optimized" construction of CCA-secure PKE from TB-ATDF given in Section 3.

Recall that this construction proceeds in two steps. First, we construct a selective-tag weakly CCA-secure PKE scheme in the sense of [?] by using the TB-ATDF as a key-encapsulation mechanism for a one-time IND-CPA secure symmetric encryptions scheme. We note that to extract enough hardcore bits from only one application of RSA, we can combine $\mathsf{II\text{-}RSA}$ with the "small-solutions" RSA problem of [?]. Furthermore, by strengthening $\mathsf{II\text{-}RSA}$ to allow e, e' to be composites such that $\gcd(e, e') = 1$ and quantifying over *all* e in the assumption, we can "heuristically" use a cryptographic hash function with 512-bit output in place of the prime sequence generator for 80-bit security.

The construction then applies the MAC-based BCHK-transform [?] to obtain a fully CCA-secure PKE scheme. The resulting scheme has ciphertexts containing only one group element and, assuming the strengthening to $\mathsf{II\text{-}RSA}$ discussed above, its encryption time is dominated by one 512-bit exponentiation. In terms of applicability, however, it is unclear if such a standard-model PKE scheme secure based on an interactive assumption about RSA (such as $\mathsf{II\text{-}RSA}$) is preferable to a random-oracle scheme based on its one-wayness (such as RSA-OAEP [?]).

Acknowledgements

We thank Mihir Bellare for helpful discussions that improved the presentation of our results. We also thank Chris Peikert, Ramarathnam Venkatesan, and the anonymous reviewers of Eurocrypt 2010 for helpful comments. Eike Kiltz was supported by the research program Sentinels. Sentinels is being financed by Technology Foundation STW, the Netherlands Organization for Scientific Research (NWO), and the Dutch Ministry of Economic Affairs. Adam O'Neill was supported in part by Alexandra Boldyreva's NSF CAREER award 0545659 and NSF Cyber Trust award 0831184 and thanks her for her support.