Instantiability of RSA-OAEP under Chosen-Plaintext Attack

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

[1] Centrum voor Wiskunde en Informatica, Amsterdam, Netherlands
kiltz@cwi.nl
[2] Georgia Institute of Technology, Atlanta, GA, USA
amoneill@cc.gatech.edu
[3] Pennsylvania State University, University Park, PA, USA
asmith@cse.psu.edu

Abstract

We show that the widely deployed RSA-OAEP encryption scheme of Bellare and Rogaway (Eurocrypt 1994), which combines RSA with two rounds of an underlying Feistel network whose hash (*i.e.*, round) functions are modeled as random oracles, meets indistinguishability under chosen-plaintext attack (IND-CPA) in the *standard model* based on simple, non-interactive, and non-interdependent assumptions on RSA and the hash functions. To prove this, we first give a result on a more general notion called "padding-based" encryption, saying that such a scheme is IND-CPA if (1) its underlying padding transform satisfies a "fooling" condition against small-range distinguishers on a class of high-entropy input distributions, and (2) its trapdoor permutation is sufficiently *lossy* as defined by Peikert and Waters (STOC 2008). We then show that the first round of OAEP satifies condition (1) if its hash function is t-wise independent for appopriate t and that RSA satisfies condition (2) under the Φ-Hiding Assumption of Cachin *et al.* (Eurocrypt 1999).

This appears to be the first non-trivial *positive* result about the instantiability of RSA-OAEP. In particular, it increases our confidence that chosen-plaintext attacks are unlikely to be found against the scheme. In contrast, RSA-OAEP's predecessor in PKCS #1 v1.5 was shown to be vulnerable to such attacks by Coron *et al.* (Eurocrypt 2000).

1 Introduction

The RSA-OAEP encryption scheme was designed by Bellare and Rogaway [5] as a drop-in replacement for RSA PKCS #1 v1.5 [37] with provable security guarantees. In particular, it follows the same paradigm as RSA PKCS #1 v1.5 in that it encrypts a message of less than k bits to a k-bit ciphertext (where k is the modulus size) by first applying a fast, randomized, and invertible "padding transform" to the message before applying RSA. In the case of RSA-OAEP, the underlying padding transform (which is itself called 'OAEP'[1]) embeds a message m and random coins r as $s\|(H(s) \oplus r)$ where '$\|$' denotes concatenation,

[1] We often use the same terminology for 'f-OAEP,' which refers to OAEP using an abstract TDP f, with the meaning hopefully clear from context.

$s = (m\|0^{k_1}) \oplus G(r)$ for some parameter k_1, and G and H are hash functions (see Figure 1 on p. 11). In contrast, PKCS #1 v1.5 essentially just concatenates m with r.

RSA-OAEP was designed using the random oracle (RO) methodology [4]. This means that, for the security analysis, its hash functions are modeled as independent truly random functions, available as oracles to all parties. When the scheme is implemented in practice, they are heuristically "instantiated" in certain ways using a cryptographic hash function like SHA1. A cryptographic hash function is certainly not random (it has a short public description), but schemes designed using this methodology are hoped to be secure. Unfortunately, a series of works, starting with the seminal paper of Canetti *et al.* [16] showed that there are schemes secure in the RO model that are insecure under *every* instantiation of the oracle; such RO model schemes are called *uninstantiable*. Thus, to gain confidence in an RO model scheme, we should show that it is not uninstantiable, *i.e.*, that it admits a secure instantiation by an efficiently computable function under well-defined assumptions. Then, when we instantiate the scheme, we know that our goal is at least plausible. This is especially important for a scheme such as RSA-OAEP, which is by now widely standardized and deployed.

Yet, while RO model schemes continue to be proposed, few have been shown to be instantiable. In particular, we are not aware of *any* result showing instantiability of RSA-OAEP, even under a relatively modest security model. In fact, the scheme has come under criticism lately due to several works (discussed in Section 1.2) showing the impossibility of certain types of instantiations under chosen-ciphertext attack (IND-CCA). Fortunately, we bring some good news: We give reasonable assumptions under which RSA-OAEP is secure against *chosen-plaintext attack* (IND-CPA). We believe this is an important step towards a better understanding of the scheme's security.

1.1 Our Contributions

Our result on the instantiability of RSA-OAEP is obtained via three steps or other results. (These other results may also be of independent interest.) First, we show a general result on the instantiability of "padding-based encryption," of which f-OAEP is a special case, under the assumption that the underlying padding transform is what we call a *fooling extractor* and the trapdoor permutation is sufficiently *lossy* [36]. We then show that OAEP and RSA satisfy the respective conditions.

Padding-based encryption without ROs. Our first result is a general theorem about *padding-based encryption* (PBE), a notion formalized recently by Kiltz and Pietrzak [29].[2] PBE generalizes the design methology of PKCS #1 and RSA-OAEP we already mentioned. Namely, we start with a k-bit to k-bit trapdoor permutation (TDP) that satisfies a weak security notion like one-wayness.

[2] Such schemes were called "simple embedding schemes" by Bellare and Rogaway [5], who discussed them only on an intuitive level.

To "upgrade" the TDP to an encryption scheme satisfying a strong security notion like IND-CPA, we design an invertible "padding transform" which embeds a plaintext and random coins into a k-bit string, to which we then apply the TDP. This methodology is quite natural and has long been prevalent in practice, motivating the design of OAEP and later schemes such as SAEP [9] and PSS-E [20]. The latter were all designed and analyzed in the RO model.

We show that the RO model is *unnecessary* in the design and analysis of IND-CPA secure PBE. To do so, we formulate an interesting connection between PBE and a new notion we call "fooling extractor for small-range distinguishers" or just "fooling extractor." Intuitively, a fooling extractor transforms a high-entropy source into something that "looks random" to any function (or distinguisher) with a *small range*.[3] Our result says that if the underlying padding transform of a PBE scheme is a fooling extractor for all sources of the form (m, R) where m is a plaintext and R is the random coins (which we call "encryption sources") and its TDP is *lossy* as defined by Peikert and Waters [36] then the PBE scheme is IND-CPA. We call such padding transforms "encryption-compatible."

OAEP FOOLS SMALL-RANGE DISTINGUISHERS. Our second result says that the OAEP padding transform is encryption-compatible as we defined it above if the hash function G is t-wise independent for appropriate t (essentially, proportional to the allowed message length, where the latter is determined by how large an output range of the distinguisher should be tolerated in the definition of encryption-compatibility). Note that no restriction is put on hash function H; in particular, neither hash function is modeled as a RO.

The inspiration for our proof comes from the "Crooked" Leftover Hash Lemma (LHL) of Dodis and Smith [22] (see [6] for a simpler proof of the latter). Qualitatively, the Crooked LHL says that $K, f(\Pi(K, X))$ looks like $K, f(U)$ for any small-range function f, pairwise-independent function Π keyed by K, and high-entropy source X; in our terminology, this says that a pairwise-independent function is a fooling extractor for such X. In our application, we might naïvely view Π as the OAEP. There are two problems with this. First, OAEP is *not* pairwise independent, even in the RO model. Second, showing that OAEP is encryption-compatible entails showing it fools f on *all* encryption sources simultaneously, whereas the lemma pertains to a *fixed* source. To solve the first problem, we show that the lemma can be strengthened to say that $K, f(X, \Pi(K, X))$ looks like $K, f(X, U)$; *i.e.*, that $\Pi(K, X)$ looks random to f *even given* X. Then, we view X as the random coins in OAEP and Π as the hash function G; we can conclude that OAEP is a fooling extractor for a *fixed* encryption source (m, R) (note that our analysis does not use any properties of H—the only fact we use about the second Feistel round is that it is invertible). To solve the second problem, we extend an idea of Trevisan and Vadhan [42] to our setting and show that if G is in fact t-*wise* independent for large enough t, the error probability for a particular encryption source is so small that we can take a union bound and conclude that OAEP is a fooling extractor on all of them, as required.

[3] In the formal defintion there is also an "outer" distinguisher who gets the extractor seed; see Section 3 for details.

LOSSINESS OF RSA. To instantiate RSA-OAEP, it remains to show lossiness of RSA. Our final result is that RSA is indeed lossy under reasonable assumptions. Intuitively, lossiness [36] means that there is an alternative, "lossy" key generation algorithm that outputs a public key indistinguishable from a normal one, but which induces a small-range (uninvertible) function. We first show lossiness of RSA under the Φ-Hiding Assumption (ΦA) of Cachin, Micali, and Stadler [13]. ΦA has been used as the basis for a number of efficient protocols, *e.g.*, [13,12,24,25]. ΦA states roughly that given an RSA modulus $N = pq$, it is hard to distinguish primes that divide $\phi(N) = (p-1)(q-1)$ from those that do not. Normal RSA parameters (N, e) are such that $\gcd(e, \phi(N) = 1$. Under ΦA, we may alternatively choose (N', e) such that e *divides* $p-1$. The range of the RSA function is then reduced by a factor $1/e$. To resist known attacks, we can take the bit-length of e up to almost $1/4$ that of N, giving RSA lossiness of almost $k/4$ bits, where k is the modulus length.[4]

We then observe that for small e lossiness may be amplified for a fixed modulus length by considering *multi-prime* RSA where $N = p_1 \cdots p_m$ for $m \geq 2$, and in the lossy case choosing (N', e) such that e divides p_i for *all* $1 \leq i \leq m-1$; the range of the RSA function is then reduced by a factor $1/e^{m-1}$. (The maximum bit-length of e in this case to avoid known attacks is roughly $k(1/m - 2/m^2)$ where k is the modulus length, so for a fixed modulus size we gain in lossiness only for small e.) If we assume such multi-prime RSA moduli are indistinguishable from two-prime ones, we can achieve such lossiness in the case of standard (two-prime) RSA as well.

IMPLICATIONS FOR RSA-OAEP. Combining the above implies that RSA-OAEP is IND-CPA in the standard model under (rather surprisingly) simple, non-interactive, and non-interdependent assumptions on RSA and the hash functions. The parameters for RSA-OAEP supported by our proofs are discussed in Section 6. While they are considerably worse than what is expected in practice, we view the upshot of our results not as the concrete parameters they support, but rather that they increase the theoretical backing for the scheme's security at a more qualitative level, showing it can be instantiated at least for larger parameters. In particular, our results give us greater confidence that chosen-plaintext attacks are unlikely to be found against the scheme; such attacks are known against the predecessor of RSA-OAEP in PKCS #1 v1.5 [19]. That said, we strongly encourage further research to try to improve the concrete parameters.

Moreover, our analysis brings to light to some simple modifications that may increase the scheme's security. The first is to key the hash function G. Although our results have some interpretation in the case that G is a fixed function (see below), it may be preferable for G to have an explicit, randomly selected key. It is in an interesting open question whether our proof can be extended to function families that use shorter keys. The second possible modification is to increase the length of the randomness versus that of the redundancy in the message

[4] We remark that the recent attacks on ΦA [40] are for moduli of a special form that does not include RSA.

when encrypting short messages under RSA-OAEP. Of course, we suggest these modifications only in cases where they do not impact efficiency too severely.

Using unkeyed hash functions. Formally, our results assume G is randomly chosen from a large family (*i.e.*, it is a keyed hash function). However, our analysis actually shows that *almost every* function (*i.e.*, all but a negligible fraction) from the family yields a secure instantiation; we just do not know an explicit member that works. In other words, it is not strictly necessary that G be randomly chosen. When G is instantiated in practice using a cryptographic hash function, it is plausible that the resulting instantiation is secure.

Chosen-ciphertext security. Any extension of our results to CCA security must get around the recent negative results of Kiltz and Pietrzak [29] (which we discuss in more detail below). We outline some possible approaches in the full version [27].

1.2 Related Work

Security of OAEP in the RO model. In their original paper [5], Bellare and Rogaway showed that OAEP is IND-CPA assuming the TDP is one-way. They further showed it achieves a notion they called "plaintext awareness." Subsequently, Shoup [41] observed that the latter notion is too weak to imply security against chosen-ciphertext attacks, and in fact there is no black-box proof of IND-CCA security of OAEP based on one-wayness of the TDP. Fortunately, Fujisaki *et al.* [23] proved that OAEP is nevertheless IND-CCA assuming so-called "partial-domain" one-wayness, and that partial-domain one-wayness and (standard) one-wayness of RSA are equivalent.

Security of OAEP without ROs. Results on instantiability of OAEP have so far mainly been negative. Boldyreva and Fischlin [7] showed that (contrary to a conjecture of Canetti [14]) one cannot securely instantiate even *one* of the two hash functions (while still modeling the other as a RO) of OAEP under IND-CCA by a "perfectly one-way" hash function [14,17] if one assumes only that f is partial-domain one-way. Brown [10] and Paillier and Villar [34] later showed that there are no "key-preserving" black-box proofs of IND-CCA security of RSA-OAEP based on one-wayness of RSA. Recently, Kiltz and Pietrzak [29] (building on the earlier work of Dodis *et al.* [21] in the signature context) generalized these results and showed that there is no black-box proof of IND-CCA (or even NM-CPA) security of OAEP based on any property of the TDP satisfied by an *ideal* (truly random) permutation.[5] In fact, their result can be extended to rule out a black-box proof of NM-CPA security of OAEP assuming the TDP is lossy [30], so our results are in some sense *optimal* given our assumptions.

Instantiations of related schemes. A positive instantiation result about a variant of OAEP called OAEP++ [26] (where part of the transform is output

[5] Note, however, that their result does not rule out such a proof based on other properties of the TDP, non-black-box assumptions on the hash functions, or in the case of a specific TDP like RSA.

in the clear) was obtained by Boldyreva and Fischlin in [8]. They showed an instantiation that achieves (some weak form of) non-malleability under chosen-plaintext attacks (NM-CPA) for random messages, assuming the existence of non-malleable pseudorandom generators (NM-PRGs).[6] We note that the approach of trying to obtain positive results for instantiations under security notions weaker than IND-CCA originates from their work, and the authors explicitly ask whether OAEP can be shown IND-CPA in the standard model based on reasonable assumptions on the TDP and hash functions.

Another line of work has looked at instantiating other RO model schemes related at least in spirit to OAEP. Canetti [14] showed that the IND-CPA scheme in [4] can be instantiated using (a strong form of) perfectly-one way probabilistic hash functions. More recently, the works of Canetti and Dakdouk [15], Pandey *et al.* [35], and Boldyreva *et al.* [11] obtained (partial) instantiations of the earlier IND-CCA scheme of [4]. Hofheinz and Kiltz [28] recently showed an IND-CCA secure instantiation of a variant the DHIES scheme of [1].

2 Preliminaries

Notation and conventions. For a probabilistic algorithm A, by $y \xleftarrow{\$} A(x)$ we mean that A is executed on input x and the output is assigned to y, whereas if S is a finite set then by $s \xleftarrow{\$} S$ we mean that s is assigned a uniformly random element of S. We sometimes use $y \leftarrow A(x; \mathsf{Coins})$ to make A's random coins explicit. We denote by $\Pr[A(x) \Rightarrow y : \ldots]$ the probability that A outputs y on input x when x is sampled according to the elided experiment. Unless otherwise specified, an algorithm may be probabilistic and its running-time includes that of any overlying experiment. We denote by 1^k the unary encoding of the security parameter k. We sometimes surpress dependence on k for readability. For $i \in \mathbb{N}$ we denote by $\{0,1\}^i$ the set of all binary strings of length i. If s is a string then $|s|$ denotes its length in bits, whereas if S is a set then $|S|$ denotes its cardinality. By '$\|$' we denote string concatenation. All logarithms are base 2.

Basic Definitions. Writing $P_X(x)$ for the probability that a random variable X puts on x, the *statistical distance* between random variables X and Y with the same range is given by $\Delta(X,Y) = \frac{1}{2}\sum_x |P_X(x) - P_Y(x)|$. If $\Delta(X,Y)$ is at most ε then we say X, Y are ε-*close* and write $X \approx_\varepsilon Y$. The *min-entropy* of X is $\mathrm{H}_\infty(X) = -\log(\max_x P_X(x))$. A random variable X over $\{0,1\}^n$ is called a (n,ℓ)-*source* if $\mathrm{H}_\infty(X) \geq \ell$. Let $f : A \to B$ be a function. We denote by $R(f)$ the *range* of f, *i.e.*, $\{b \in B \mid \exists a \in A, f(a) = b\}$. We call $|R(f)|$ the *range-size* of f. We call f *regular* if each pre-image set is the same size, *i.e.*, $|\{x \in D \mid f(x) = y\}|$ is the same for all $y \in R$.

Public-key encryption and its security. A *public-key encryption scheme* with message-space MsgSp is a triple of algorithms $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$. The key-

[6] In particular, their security notion does *not* imply IND-CPA since they consider random messages. We also point out that it remains an open question whether NM-PRGs can be constructed.

generation algorithm $\mathcal{K}$ returns a public key pk and matching secret key sk. The encryption algorithm $\mathcal{E}$ takes pk and a plaintext m to return a ciphertext. The deterministic decryption algorithm $\mathcal{D}$ takes sk and a ciphertext c to return a plaintext. We require that for all messages $m \in \mathrm{MsgSp}$

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

is negligible.

To an encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ and an adversary $A = (A_1, A_2)$ we associate a chosen-plaintext attack experiment,

$$\begin{array}{l} \textbf{Experiment } \mathbf{Exp}^{\text{ind-cpa}}_{\Pi,A}(k) \\ \quad b \xleftarrow{\$} \{0,1\} \,;\; (pk, sk) \xleftarrow{\$} \mathcal{K}(1^k) \\ \quad (m_0, m_1, \mathrm{state}) \xleftarrow{\$} A_1(pk) \\ \quad c \xleftarrow{\$} \mathcal{E}(pk, m_b) \\ \quad d \xleftarrow{\$} B_2(pk, c, \mathrm{state}) \\ \quad \text{If } d = b \text{ then return 1 else return 0} \end{array}$$

where we require A's output to satisfy $|m_0| = |m_1|$. Define the *ind-cpa advantage* of A against Π as

$$\mathbf{Adv}^{\text{ind-cpa}}_{\Pi,A}(k) = 2 \cdot \Pr\left[\, \mathbf{Exp}^{\text{ind-cpa}}_{\Pi,A}(k) \Rightarrow 1 \,\right] - 1 \,.$$

Lossy trapdoor permutations. A *lossy trapdoor permutation (LTDP) generator* [36][7] is a pair $\mathsf{LTDP} = (\mathcal{F}, \mathcal{F}')$ of algorithms. Algorithm $\mathcal{F}$ is a usual trapdoor permutation (TDP) generator, namely it outputs a pair (f, f^{-1}) where f is a (description of a) permutation on $\{0,1\}^k$ and f^{-1} its inverse. Algorithm $\mathcal{F}'$ outputs a (description of a) function f' on $\{0,1\}^k$. We call $\mathcal{F}$ the "injective mode" and $\mathcal{F}'$ the "lossy mode" of LTDP respectively, and we call $\mathcal{F}$ "lossy" if it is the first component of some lossy TDP. For a distinguisher D, define its *ltdp-advantage* against LTDP as

$$\mathbf{Adv}^{\text{ltdp}}_{\mathsf{LTDP},D}(k) = \Pr\left[\, D(f) \Rightarrow 1 \;:\; (f, f^{-1}) \xleftarrow{\$} \mathcal{F} \,\right] - \Pr\left[\, D(f') \Rightarrow 1 \;:\; f' \xleftarrow{\$} \mathcal{F}' \,\right] \,.$$

We say LTDP has *residual leakage* s if for all f' output by $\mathcal{F}'$ we have $|R(f')| \leq 2^s$. The *lossiness* of LTDP is $\ell = k - s$.

t-wise independent hashing. Let $H \colon \mathcal{K} \times D \to R$ be a hash function. We say that H is *t-wise independent* if for all distinct $x_1, \ldots, x_t \in D$ and all $y_1, \ldots, y_t \in R$

$$\Pr\left[\, H(K, x_1) = y_1 \;\wedge\; \ldots \;\wedge\; H(K, x_t) = y_t : \; K \xleftarrow{\$} \mathcal{K} \,\right] \;=\; \frac{1}{|R|^t} \,.$$

In other words, $H(K, x_1), \ldots, H(K, x_t)$ are all uniformly and independently random.

[7] We note that [36] actually defines lossy trapdoor *functions*, but the extension to permutations is straightforward.

3 Padding-Based Encryption from Lossy TDP + Fooling Extractor

In this section, we show a general result on how to build IND-CPA secure padding-based encryption (PBE) without using random oracles, by combining a lossy TDP with a "fooling extractor" for small-range distinguishers.

3.1 Background and Tools

We first provide the definitions relevant to our result.

PADDING-BASED ENCRYPTION. The idea behind padding-based encryption (PBE) is as follows: We start with a k-bit to k-bit trapdoor permutation (*e.g.*, RSA) and wish to build a secure encryption scheme. As in [5], we are interested in encrypting messages of less than k bits to ciphertexts of length k. It is well-known that we cannot simply encrypt messages under the TDP directly to achieve strong security. So, in a PBE scheme we "upgrade" the TDP by first applying a randomized and invertible "padding transform" to a message prior to encryption.

Our definition of PBE largely follows the recent formalization in [29]. Let k, μ, ρ be three integers such that $\mu + \rho \leq k$. A *padding transform* $(\pi, \hat{\pi})$ consists of two mappings $\pi : \{0,1\}^{\mu+\rho} \to \{0,1\}^k$ and $\hat{\pi} : \{0,1\}^k \to \{0,1\}^\mu \cup \{\perp\}$ such that π is injective and the following consistency requirement is fulfilled:

$$\forall m \in \{0,1\}^\mu, r \in \{0,1\}^\rho \ : \quad \hat{\pi}(\pi(m \,\|\, r)) = m \ .$$

A *padding transform generator* is an algorithm Π that on input 1^k outputs a (description of a) padding transform $(\pi, \hat{\pi})$. Let $\mathcal{F}$ be a k-bit trapdoor permutation generator and Π be a padding transform generator. Define the associated *padding-based encryption scheme* $\mathcal{AE}_\Pi[\mathcal{F}] = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ with message-space $\{0,1\}^\mu$ by

$\mathbf{Alg}\ \mathcal{K}(1^k)$	$\mathbf{Alg}\ \mathcal{E}((\boldsymbol{\pi}, f), m)$	$\mathbf{Alg}\ \mathcal{D}((\boldsymbol{\pi}, f^{-1}), y)$
$(\pi, \hat{\pi}) \xleftarrow{\$} \Pi(1^k)$	$r \xleftarrow{\$} \{0,1\}^\rho \,;\, x \leftarrow \pi(m \| r)$	$x \leftarrow f^{-1}(y)$
$\boldsymbol{\pi} \leftarrow (\pi, \hat{\pi})$	$y \leftarrow f(x)$	$m \leftarrow \hat{\pi}(x)$
$(f, f^{-1}) \xleftarrow{\$} \mathcal{F}(1^k)$	Return y	Return m
Return $((\boldsymbol{\pi}, f), (\boldsymbol{\pi}, f^{-1}))$		

Padding-based encryption schemes have long been prevalent in practice, for example PKCS #1 [37]. While OAEP [5] is the best-known, the notion also captures later schemes such as SAEP [9] and PSS-E [20].

FOOLING EXTRACTORS. We define a new notion that we call "fooling extractor for small-range distinguishers" or just "fooling extractor." Intuitively, fooling extractors are a type of randomness extractor that "fools" distinguishers with small-range output. We give some more intuition after the formal definition.

Definition 1. *Let* $\mathsf{FExt}\colon \{0,1\}^c \times \{0,1\}^n \to \{0,1\}^k$ *be a function and let* $\mathcal{X} = \{X_1, \ldots, X_q\}$ *be a class of* n*-bit sources. We say that* FExt fools range-2^s distinguishers on $\mathcal{X}$ with probability $1 - \varepsilon$ *(or is an* (s, ε)*-fooling extractor for* $\mathcal{X}$*) if*

for all functions f on $\{0,1\}^k$ with range-size at most 2^s and all $1 \le i \le q$:

$$(K, f(\mathsf{FExt}(K, X_i)) \approx_\varepsilon (K, f(U)) \ ,$$

where K is uniform on $\{0,1\}^c$ and U is uniform and independent on $\{0,1\}^n$. (Here K is the key *or* seed *of* FExt*.) For example, one is often interested in the class $\mathcal{X}_{n,\ell}$ consisting of all (n,ℓ)-sources X. As a strengthening of the above, we say that* FExt simultaneously *fools range-2^s distinguishers on $\mathcal{X}$ with probability $1-\varepsilon$ (or is a simultaneous (s,ε)-fooling extractor for $\mathcal{X}$) if for all functions f on $\{0,1\}^k$ with range-size at most 2^s:*

$$\mathop{\mathbf{E}}_{k \xleftarrow{\$} \{0,1\}^c} \left[\max_{1 \le i \le q} \Delta\Big(f(\mathsf{FExt}(k, X_i)) \ , \ f(U) \Big) \right] \le \varepsilon \ .$$

As a useful special case, we say that FExt *fools* regular *range-2^s distinguishers on $\mathcal{X}$ with probability $1-\varepsilon$ (or is a regular (s,ε)-fooling extractor for $\mathcal{X}$) if we quantify only over* regular *f in the definition. A* simultaneous regular *(s,ε)-fooling extractor for $\mathcal{X}$ is defined analogously.*

Intuitively, one can think of the definition of a fooling extractor as involving a two-stage distinguisher. The first stage is represented by the function f, which takes as input $\mathsf{FExt}(K, X_i)$. The second stage is represented only implicitly, and takes as input $f(\mathsf{FExt}(K, X_i))$ and K. While the intuition given prior to the definition captures only the first stage, the second stage is crucial for the definition to be meaningful. Indeed, just asking that $f(\mathsf{FExt}(K, X_i))$ be indistinguishable from $f(U)$ for all small-range functions f is equivalent to asking only that $\mathsf{FExt}(K, X_i)$ be indistinguishable from U. This latter requirement is trivial to achieve–for example, by using K as a one-time pad.

We note that the concept of fooling extractors was implicit in the work of Dodis and Smith [22] on error-correction without leaking partial information, whose "Crooked" Leftover Hash Lemma establishes in our language that a pairwise-independent function is a (s,ε)-fooling extractor for every singleton (n,ℓ)-source X where $s \le \ell - 2\log(1/\varepsilon) + 2$.

3.2 The Result

To state our result, we first formalize the concept of *encryption-compatible* padding transforms.

Definition 2. *Let Π be a padding transform generator whose coins are drawn from* Coins*. Define the function $h_\Pi : \mathsf{Coins} \times \{0,1\}^{\mu+\rho} \to \{0,1\}^k$ by $h(c, m\|r) = \pi(m\|r)$ for all $c \in \mathsf{Coins}, m \in \{0,1\}^\mu, r \in \{0,1\}^\rho$, where $(\pi, \hat{\pi}) \leftarrow \Pi(1^k; \mathsf{Coins})$. We say that Π is* (s,ε)-encryption-compatible *if h_Π as above is a simultaneous (s,ε)-fooling extractor for the class $\mathcal{X}_\Pi$ of sources of the form (m,R), where $m \in \{0,1\}^\mu$ is fixed and $R \in \{0,1\}^\rho$ is uniformly random. (Note that the class $\mathcal{X}_\Pi$ contains 2^μ distinct $(\mu+\rho)$-bit sources.) We call $\mathcal{X}_\Pi$ the class of* encryption sources *associated to Π. A* regular *(s,ε)-encryption-compatible padding transform generator is defined analogously.*

Theorem 1. *Let* $\mathsf{LTDP} = (\mathcal{F}, \mathcal{F}')$ *be an LTDP with residual leakage* s*, and let* Π *be an* (s, ε)*-encryption-compatible padding transform generator. Then for any IND-CPA adversary* A *against* $\mathcal{AE}_{\Pi}[\mathcal{F}]$ *there is a adversary* D *against* LTDP *such that for all* $k \in \mathbb{N}$

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathcal{AE},A}(k) \;\leq\; \mathbf{Adv}^{\text{ltdp}}_{\mathsf{LTDP},D}(k) + \varepsilon \;.$$

Furthermore, the running-time of D *is the time to run* A*.*

Remark 1. The analogous result to the above holds for regular LTDPs and regular encryption-compatible padding transforms. That is, if the LTDP is *regular* (meaning $\mathcal{F}'$ is) then it suffices to use a regular encryption-compatible padding transform to obtain the same conclusion. The latter may be easier to design or more efficient than in the general case; indeed, we get better parameters for OAEP in the regular case in Section 4. Furthermore, known examples of LTDPs (including RSA, as shown in Section 5) are regular, although some technical issues make it difficult to exploit this for RSA-OAEP; cf. Section 6.

4 OAEP as a Fooling Extractor

In this section, we show that the OAEP padding transform of Bellare and Rogaway [5] is encryption-compatible as defined in Section 3 if its initial hash function is t-wise independent for appropriate t.

4.1 OAEP

We recall the OAEP padding transform of Bellare and Rogaway [5], lifted to the "instantiated" setting where hash functions may be keyed. Let $G\colon \mathcal{K}_G \times \{0,1\}^{\rho} \to \{0,1\}^{\mu}$ and $H\colon \mathcal{K}_H \times \{0,1\}^{\mu} \to \{0,1\}^{\rho}$ be hash functions. The associated padding transform generator $\mathsf{OAEP}[G,H]$ on input 1^k returns $(\pi_{K_G,K_H}, \hat{\pi}_{K_G,K_G})$, where $K_G \xleftarrow{\$} \mathcal{K}_G(1^k)$ and $K_H \xleftarrow{\$} \mathcal{K}_H(1^k)$, defined via

Algorithm $\pi_{K_G,K_H}(m\|r)$	**Algorithm** $\hat{\pi}_{K_G,K_H}(x)$
$s \leftarrow m \oplus G(K_G, r)$	$s\|t \leftarrow x$
$t \leftarrow r \oplus H(K_H, s)$	$r \leftarrow t \oplus H(K_H, s)$
$x \leftarrow s\|t$	$m \leftarrow s \oplus G(K_G, r)$
Return x	Return m

See Figure 1 for a graphical illustration.

Remark 2. Since we mainly study IND-CPA security, for simplicity we define above the "no-redundancy" version of the OAEP, *i.e.*, corresponding to the "basic scheme" in [5]. However, our results also hold for the redundant version. Additionally, as is typical in the literature we have defined OAEP to apply the G-function to the least-significant bits of the input; in standards and implementations it is typically the most significant bits (where the order of m and r are switched). Again, we stress that our results hold in either case.

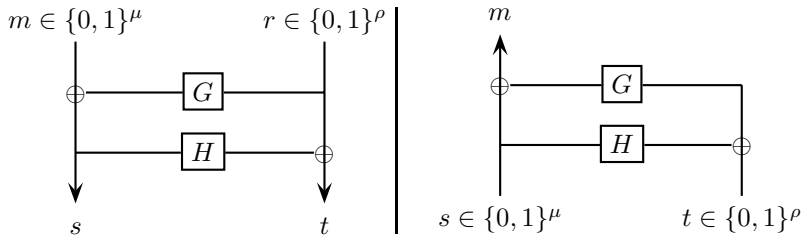

Fig. 1. Algorithms $\pi_{K_G,K_H}(m,r)$ and $\hat{\pi}_{K_G,K_H}(s,t)$ for $\mathsf{OAEP}[G,H]$.

4.2 Analysis

The following establishes that OAEP is encryption-compatible if the hash function G is t-wise independent for appropriate t. No restriction is put on the other hash function H. Indeed, our result also applies to SAEP [9] (although the latter is neither standardized nor known to provide CCA security in the RO model, except in certain cases).

Theorem 2. *Let $G\colon \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^\mu$ and $H\colon \mathcal{K}_H \times \{0,1\}^\mu \to \{0,1\}^\rho$ be hash functions, and suppose G is t-wise independent. Let $\mathsf{OAEP} = \mathsf{OAEP}[G,H]$. Then*

(1) OAEP *is (s,ε)-encryption-compatible where $\varepsilon = 2^{-u}$ for $u = \frac{t}{3t+2}(\rho - s - \log t + 2) - \frac{2(\mu+s)}{3t+2} - 1$.*

(2) OAEP *is* regular *(s,ε)-encryption-compatible where $\varepsilon = 2^{-u}$ for $u = \frac{t}{2t+2}(\rho - s - \log t + 2) - \frac{\mu+s+2}{t+1} - 1$.*

(3) *When $t = 2$, OAEP (s,ε)-encryption-compatible where $\varepsilon = 2^{-u}$ for $u = (\rho - s - 2\mu)/4 - 1$.*

Note that parts (2) and (3) capture special cases of (1) in which we get better bounds. We give a high-level idea of the proof; details are deferred to the full version [27].

The high-level idea for all three parts of the theorem is the same. Fix a lossy function f with range-size at most 2^s. We first show that for every *fixed* message $m \in \{0,1\}^\mu$, with high probability (say $1-\delta$) over the choice of the hash function G, the statistical distance between $(K_G, f(\mathsf{OAEP}(m,R)))$ and $((K_G, f(U))$ is small (say $\hat{\varepsilon}$). Namely, we first compute the *expected* statistical distance over the choice of G and then apply tail bounds. This aspect of the proof changes from part to part. For part (3) we use a strengthened version of the Crooked Leftover Hash Lemma (LHL) of [22] and Markov's inequality. For parts (1) and (2) we adapt the techniques of [42] (see also [2]) developed in the context of the standard LHL and use the tail inequality for t-wise independent random variables due to Bellare and Rompel [3]. (For part (2) this is relatively easy, but for part (1) we first apply a "balancing" lemma saying that for any non-regular f we can find a "almost-regular" function g that agrees with f on a large

fraction of its domain.) In all three parts, we can then take a union bound to show that OAEP is good for *all* messages with probability at least $1 - 2^{\mu}\delta$. This means that the statistical distance between the pair $(K_G, f(\mathsf{OAEP}(m, R)))$ and $(K_G, f(\mathsf{OAEP}(U)))$ is at most $\varepsilon = \hat{\varepsilon} + 2^{\mu}\delta$. Finally, we express δ as a function of $\hat{\varepsilon}$, and select $\hat{\varepsilon}$ to minimize this sum. Note that the entire argument works for any choice of H.

In order to get a more qualitative "feel" for the bounds in the theorem, we give the following simplification as a corollary:

Corollary 1. *Let $G\colon \mathcal{K}_G \times \{0,1\}^{\rho} \to \{0,1\}^{\mu}$ and $H\colon \mathcal{K}_H \times \{0,1\}^{\mu} \to \{0,1\}^{\rho}$ be hash functions and suppose that G is t-wise independent for $t \geq 3\frac{\mu+s}{\rho-s}$. Then $\mathsf{OAEP}[G, H]$ is (s, ε)-encryption-compatible where $\varepsilon = \exp(-c(\rho - s - \log t))$ for a constant $c > 0$.*

In particular, $c \approx 1/2$ for regular functions. For such a function, if $\rho - s$ is at least 180 then ε is roughly 2^{-80} for $t = 10$ and message lengths $\mu \leq 2^{15}$ (which for practical purposes does not restrict the message-space). Applying Theorem 1, we see that if G is 10-wise independent and the number of random bits used in OAEP is at least 180 bits larger than the residual lossiness of the TDP, then the security of OAEP is tightly related to that of the lossy TDP.

Remark 3. To show security of OAEP against what we call *key-independent* chosen-plaintext attack, it suffices to argue that $\mathsf{OAEP}[G, H]$ is a fooling extractor for any *fixed* encryption source $X = (m, R)$ where $m \in \{0,1\}^{\mu}$. The latter holds for any $\varepsilon > 0$ and $s \leq \rho - 2\log(1/\varepsilon) + 2$ assuming G is only pairwise-independent (*i.e.*, $t = 2$). See the full version [27] for details.

5 Lossiness of RSA

In this section, we show that the RSA trapdoor permutation is lossy under reasonable assumptions. In particular, we show that, for large enough encryption exponent e, RSA is considerably lossy under the Φ-Hiding Assumption of [13]. We then show that by generalizing this assumption to multi-prime RSA we can get even more lossiness. Finally, we propose a "Two-Or-m-Primes" Assumption that, when combined with the former, amplifies the lossiness of standard (two-prime) RSA for small e.

5.1 Background on RSA and Notation

We denote by $\mathcal{RSA}_k$ the set of all tuples (N, p, q) such that $N = pq$ is the product of two distinct $k/2$-bit primes. Such an N is called an *RSA modulus*. By $(N, p, q) \xleftarrow{\$} \mathcal{RSA}_k$ we mean that (N, p, q) is sampled according to the uniform distribution on $\mathcal{RSA}_k$. An *RSA TDP generator* [38] is an algorithm $\mathcal{F}$ that returns $(N, e), (N, d)$, where N is an RSA modulus and $ed \equiv 1 \pmod{\phi(N)}$. (Here $\phi(\cdot)$ denotes Euler's totient function, so in particular $\phi(N) = (p-1)(q-1)$.) The tuple (N, e) defines the permutation on $\mathbb{Z}_N^*$ given by $f(x) = x^e \bmod N$,

and similarly (N, d) defines its inverse. We say that a lossy TDP generator $\mathsf{LTDP} = (\mathcal{F}, \mathcal{F}')$ is an RSA LTDP if $\mathcal{F}$ is an RSA TDP generator.

To define the Φ-Hiding Assumption and later some extensions of it, the following notation is also useful. For $i \in \mathbb{N}$ we denote by $\mathcal{P}_i$ the set of all i-bit primes. Let R be a relation on p and q. By $\mathcal{RSA}_k[R]$ we denote the subset of $\mathcal{RSA}_k$ for that the relation R holds on p and q. For example, let e be a prime. Then $\mathcal{RSA}_k[p = 1 \bmod e]$ is the set of all (N, p, q), where where $N = pq$ is the product of two distinct $k/2$-bit primes p, q and $p = 1 \bmod e$. That is, the relation $R(p, q)$ is true if $p = 1 \bmod e$ and q is arbitrary. By $(N, p, q) \xleftarrow{\$} \mathcal{RSA}_k[R]$ we mean that (N, p, q) is sampled according to the uniform distribution on $\mathcal{RSA}_k[R]$.

5.2 RSA Lossy TDP from Φ-Hiding

Φ-Hiding Assumption (ΦA). We recall the Φ-Hiding Assumption of [13]. For an RSA modulus N, we say that N *ϕ-hides* a prime e if $e \mid \phi(N)$. Intuitively, the assumption is that, given RSA modulus N, it is hard to distinguish primes which are ϕ-hidden by N from those that are not. Formally, let $0 < c < 1/2$ be a (public) constant determined later. Consider the following two distributions:

$$\mathcal{R}_1 = \{(e, N) \ : \ e, e' \xleftarrow{\$} \mathcal{P}_{ck} \ ; \ (N, p, q) \xleftarrow{\$} \mathcal{RSA}_k[p = 1 \bmod e']\}$$
$$\mathcal{L}_1 = \{(e, N) \ : \ e \xleftarrow{\$} \mathcal{P}_{ck} \ ; \ (N, p, q) \xleftarrow{\$} \mathcal{RSA}_k[p = 1 \bmod e])\} \ .$$

To a distinguisher D we associate its ΦA *advantage* defined as

$$\mathbf{Adv}^{\Phi\mathrm{A}}_{c,D}(k) \ = \ \Pr[\, D(\mathcal{R}_1) \Rightarrow 1 \,] - \Pr[\, D(\mathcal{L}_1) \Rightarrow 1 \,] \ .$$

As shown in [13], distributions $\mathcal{R}_1, \mathcal{L}_1$ can be sampled efficiently assuming the widely-accepted Extended Riemann Hypothesis.[8]

RSA LTDP from ΦA. We construct an RSA LTDP based on ΦA. In injective mode the public key is (N, e) where e is not ϕ-hidden by N, whereas in lossy mode it is. Namely, define $\mathsf{LTDP}_1 = (\mathcal{F}_1, \mathcal{F}'_1)$ as follows:

Algorithm $\mathcal{F}_1$	**Algorithm** $\mathcal{F}'_1$
$e, e' \xleftarrow{\$} \mathcal{P}_{ck}$	$e \xleftarrow{\$} \mathcal{P}_{ck}$
$(N, p, q) \xleftarrow{\$} \mathcal{RSA}_k[p = 1 \bmod e', p]$	$(N, p, q) \xleftarrow{\$} \mathcal{RSA}_k[p = 1 \bmod e]$
If $\gcd(e, \phi(N)) \neq 1$ then return $\perp$	Return (N, e)
$d \leftarrow e^{-1} \bmod \phi(N)$	
Return $((N, e), (N, d))$	

The fact that algorithm $\mathcal{F}_1$ has only a negligible probability of failure (returning $\perp$) follows from the fact that $\phi(N)$ can have only a constant number of prime factors of length ck and Bertrand's Postulate.

[8] This is done by choosing a uniform $(1/2 - c)k$-bit number x until $p = xe + 1$ is a prime.

Proposition 1. *Suppose there is a distinguisher D against LTDP_1. Then there is a distinguisher D' such that for all $k \in \mathbb{N}$*

$$\mathbf{Adv}^{\mathrm{ltdp}}_{\mathsf{LTDP}_1,D}(k) \;\leq\; 2 \cdot \mathbf{Adv}^{\Phi\mathrm{A}}_{c,D'}(k)\;.$$

Furthermore, the running-time of D' is that of D. LTDP_1 has lossiness ck.

Remark 4. From a practical perspective, a drawback of LTDP_1 is that $\mathcal{F}_1$ chooses $N = pq$ in a non-standard way, so that it hides a prime of the same length as e. Moreover, for small values of e it returns $\perp$ with high probability. This is done for consistency with how [13] formulated ΦA. But, to address this, we also propose what we call the *Enhanced* ΦA (EΦA), which says that N generated in the non-standard way (*i.e.*, by $\mathcal{F}_1$) is indistinguishable from one chosen at random subject to $\gcd(e, \phi(N)) = 1$.[9] We conjecture that EΦA holds for all values of c that ΦA does. Details are given in the full version [27]. An analogous enhancement pertains to later extensions of ΦA.

Parameters for LTDP_1. When e is too large, ΦA can be broken by using Coppersmith's method for finding small roots of a univariate modulo an unknown divisor of N [18,32]. (No other attack on ΦA here is known.) Namely, consider the polynomial $r(x) = ex + 1 \bmod p$. Coppersmith's method allows us to find all roots of r smaller than $N^{1/4}$, and thus factor N, in lossy mode in polynomial time if $c \geq 1/4$. (This is essentially the "factoring with high bits known" attack.) More specifically, applying [32, Theorem 1], N can be factored in time $O(N^{\varepsilon})$ if $c = 1/4 - \varepsilon$ (*i.e.*, $\log e \geq k(1/4 - \varepsilon)$). For example, with modulus size $k = 2048$, for about 80-bit security in lossy mode we set $\varepsilon = .04$ (to enforce $k\varepsilon \geq 80$). The lossiness of LTDP_1 is then 432 bits according to Proposition 1. A similar calculation shows that for a modulus of size 1024 (resp., 3072) the lossiness of LTDP_1 we get is 176 (resp., 688) bits.

5.3 RSA Lossy TDP from Multi-Prime Φ-Hiding

Multi-prime RSA (according to [31] the earliest reference is [39]) is a generalization of RSA to moduli $N = p_1 \cdots p_m$ of length k with $m \geq 2$ prime factors of equal bit-length. Multi-prime RSA is of interest to practitioners since it allows to speed up decryption and is included in RSA PKCS #1 v2.1. We are interested in it here because for it we can show greater lossiness and even with smaller encryption exponent e.

Notation and terminology. Let $m \geq 2$ be fixed. We denote by $\mathcal{MRSA}_k$ the set of all tuples $(N, p_1, \ldots, p_m)$, where $N = p_1 \cdots p_m$ is the product of distinct k/m-bit primes. Such an N is called an *m-prime RSA modulus*. By $(N, p_1, \ldots, p_m) \xleftarrow{\$} \mathcal{MRSA}_k$ we mean that $(N, p_1, \ldots, p_m)$ is sampled according

[9] Additionally, in practice the encryption exponent e is usually fixed. This can be addressed by parameterizing EΦA by a fixed e instead of choosing it at random. Note that for $e = 3$ one should make both $e \mid p - 1$ and $e \mid q - 1$ in the lossy case (otherwise the assumption is false; cf. [13, Remark 2, p. 6]).

to the uniform distribution on $\mathcal{MRSA}_k$. The rest of the notation and terminology of Section 5 is extended to the multi-prime setting in the obvious way.

MULTI Φ-HIDING ASSUMPTION. For an m-prime RSA modulus N, let us say that N *$m\phi$-hides* a prime e if $e \mid p_i - 1$ for all $1 \le i \le m-1$.Intuitively, the assumption is that, given such N, it is hard to distinguish primes which are $m\phi$-hidden by N from those that do not divide $p_i - 1$ for any $1 \le i \le m$. Formally, let $m = m(k) \ge 2$ be a polynomial and let $c = c(k)$ be an inverse polynomial determined later. Consider the following two distributions:

$$\mathcal{R}_2 = \{(e, N) \ : \ e, e' \xleftarrow{\$} \mathcal{P}_{ck} \ ; \ (N, p_1, \ldots, p_t) \xleftarrow{\$} \mathcal{MRSA}_k[p_{i \le m-1} = 1 \bmod e']\}$$
$$\mathcal{L}_2 = \{(e, N) \ : \ e \xleftarrow{\$} \mathcal{P}_{ck} \ ; \ (N, p_1, \ldots, p_t) \xleftarrow{\$} \mathcal{MRSA}_k[p_{i \le m-1} = 1 \bmod e]\} \ .$$

Above and in what follows, by $p_{i \le m-1} = 1 \bmod e$ we mean that $p_i = 1 \bmod e$ for all $1 \le i \le m-1$. To a distinguisher D we associate its *MΦA advantage* defined as

$$\mathbf{Adv}^{\mathrm{M}\Phi\mathrm{A}}_{m,c,D}(k) \ = \ \Pr[\, D(\mathcal{R}_2) \Rightarrow 1 \,] - \Pr[\, D(\mathcal{L}_2) \Rightarrow 1 \,] \ .$$

As before, distributions $\mathcal{R}_2, \mathcal{L}_2$ can be sampled efficiently assuming the widely-accepted Extended Riemann Hypothesis.

Note that if we had required that in the lossy case $N = p_1 \cdots p_m$ is such that $e \mid p_i$ for all $1 \le i \le m$, then in this case we would always have $N = 1 \bmod e$. But in the injective case $N \bmod e$ is random, which would lead to a trivial distinguishing algorithm. This explains why we do not impose $e \mid p_m$ in the lossy case above.

MULTI-PRIME RSA LTDP FROM MΦA. We construct a multi-prime RSA LTDP based on MΦA having lossiness $(m-1)\log e$, where in lossy mode N $m\phi$-hides e. Namely, define $\mathsf{LTDP}_2 = (\mathcal{F}_2, \mathcal{F}'_2)$ as follows:

Algorithm $\mathcal{F}_2$	**Algorithm** $\mathcal{F}'_2$
$e, e' \xleftarrow{\$} \mathcal{P}_{ck}$	$e \xleftarrow{\$} \mathcal{P}_{ck}$
$(N, p_1, \ldots, p_m)$	$(N, p_1, \ldots, p_m)$
$\quad \xleftarrow{\$} \mathcal{MRSA}_k[p_{i \le m-1} = 1 \bmod e']$	$\quad \xleftarrow{\$} \mathcal{MRSA}_k[p_{i \le m-1} = 1 \bmod e]$
If $\gcd(e, \phi(N)) \ne 1$ then Return $\perp$	Return (N, e)
$d \leftarrow e^{-1} \bmod \phi(N)$	
Else return $(N, e), (N, d)$	

Proposition 2. *Suppose there is a distinguisher D against LTDP_2. Then there is a distinguisher D' such that for all $k \in \mathbb{N}$*

$$\mathbf{Adv}^{\mathrm{ltdp}}_{\mathsf{LTDP}_2, D}(k) \ \le \ 2 \cdot \mathbf{Adv}^{\mathrm{M}\Phi\mathrm{A}}_{m,c,D'}(k) \ .$$

Furthermore, the running-time of D' is that of D. LTDP_2 has lossiness $(m-1)ck$.

PARAMETERS FOR LTDP_2. As in the case of LTDP_1, if e is too large then Coppersmith's method [18] can be used to factor N in the lossy case. But this time the

attack is more involved than "factoring with high bits known." Let us first consider $m = 3$. Consider the polynomial $r(x'_1, x'_2) = (ex'_1 + 1)(ex'_2 + 1) \bmod p_1 p_2$. Substituting $x_1 = x'_1 x'_2$ and $x_2 = x'_1 + x'_2$ gives $r(x_1, x_2) = e^2 x_1 + e x_2 + 1 \bmod p_1 p_2$. Applying [33, Theorem 3] with $\beta = 2/3$ and $\gamma = 2\delta$ tells us that we can find all roots smaller than N^δ for $\delta = (2(1 - 2/3)^{3/2})/3 \approx .12$ in polynomial time, so we require $c \leq 1/3 - .12 \approx .21$ to prevent this attack. (Note that is slightly smaller than what we would deduce from "factoring with high bits known" [32], which gives $c \leq .22$.) More specifically, for $m = 3$ we can factor N in the lossy case in time $O(N^\varepsilon)$ if $c \geq 1/3 - \delta - \varepsilon$ (*i.e.*, $\log e = k(1/3 - \delta - \varepsilon)$) with δ as above.

In the general case, we can apply [33, Theorem 4] to deduce we must require $c \leq 1/m - \delta$ where

$$\delta \;=\; \frac{2((1/m)^{(1/m)-1} - (1/m)^{m/(m-1)})}{m(m-1)} \;\leq\; \frac{2}{m(m-1)}\,.$$

Note that this is only smaller than the bound with $\delta = 1/m^2$ obtained from "factoring with high bits known" for $m \geq 5$, namely for $m = 5$ we have $\delta \approx 0.06$. (The reason we also had a better attack for $m = 3$ is that we used a specialized theorem.)

We note that this may not be the best attack possible based on Coppersmith's method (in particular the coefficients of the polynomial we use are highly correlated). It is an interesting open question whether there is a better attack. We also remark that for a *fixed* modulus length, m cannot be too large since the Elliptic Curve Method for factoring can compute a factor p_i of N faster than the Number Field Sieve one if p_i is significantly smaller than $N^{1/2}$ [31].

5.4 Small-Exponent RSA LTDP from 2-Or-m-Primes

For efficiency reasons, the public RSA exponent e is typically not chosen to be too large in practice. (For example, researchers at UC San Diego [43] observed that 99.5% of the certificates in the campus's TLS corpus had $e = 2^{16} + 1$.) Therefore, we investigate the possibility of using an additional assumption to amplify the lossiness of RSA for small e.

The high-level idea is to assume that it is hard to distinguish $N = pq$ where p, q are primes of length $k/2$ from $N = p_1 \cdots p_m$ for $m > 2$, where $p_1, \ldots, p_m$ are primes of length k/m (which we call the "2-or-m Primes" Assumption). Combined with the MΦA Assumption of Section 5.3, we obtain $(m-1)\log e$ bits of lossiness from *standard* (two-prime) RSA. Due to space constraints, details are deferred to the full version [27].

6 Instantiating RSA-OAEP

By combining the results of Section 3, Section 4, and Section 5, we obtain standard model instantiations of RSA-OAEP under chosen-plaintext attack.

REGULARITY. In particular, we would like to apply part (2) of Theorem 2 in this case, as it is not hard to see that under all of the assumptions discussed in Section 5, RSA is a *regular* lossy TDP on the domain $\mathbb{Z}_N^*$. Unfortunately, this domain is different from $\{0,1\}^{\rho+\mu}$ (identified as integers), the range of OAEP. In RSA PKCS #1 v2.1, the mismatch is handled by selecting $\rho+\mu = \lfloor \log N \rfloor - 2$, and viewing OAEP's output as an integer less than $2^{\rho+\mu} < N/4$. The problem is that in the lossy case RSA may not be regular on the subdomain $\{0, ..., 2^{\rho+\mu}-1\}$.

We can prove, in some cases, that in the lossy case RSA is *approximately* regular on this subdomain, and in those cases we obtain the better parameters given by part (2) of Theorem 2. However, here use just use the weaker parameters given by part (1) of Theorem 2. We leave a detailed discussion of approximate regularity to future work. In particular, understanding the regularity of RSA on subintervals of the domain is a first step towards improving the concrete parameters we obtain.

CONCRETE PARAMETERS. Since the results in Section 5 have several cases and the parameter settings are rather involved, we avoid stating an explicit theorem about RSA-OAEP. From part (1) of Theorem 2 one can see that for $u = 80$ bits security and assuming RSA has ℓ bits of lossiness, messages of roughly $\mu \approx \ell - 3 \cdot 80$ bits can be encrypted (for sufficintly large t). For concreteness, we give two example parameter settings. Using the Multi Φ-Hiding Assumption with $N = 1024$ bits and 3 primes, we obtain $\ell = k - s = 291$ bits of lossiness and hence can encrypt messages of length $\mu = 40$ bits (for $t \approx 400$); using the Φ-Hiding Assumption with $N = 2048$, we obtain $\ell = k - s = 430$ bits of lossiness and hence can encrypt messages of length $\mu = 160$ bits (for $t \approx 150$). We stress that while we view our results as providing important theoretical backing for the scheme at a more qualitative level, we strongly encourage further research to try to improve the concrete parameters.

Acknowledgements

We thank Mihir Bellare, Alexandra Boldyreva, Dan Brown, Yevgeniy Dodis, Jason Hinek, Arjen Lenstra, Alex May, Phil Rogaway, and the anonymous reviewers of Crypto 2010 for helpful comments. In particular, we thank Dan for reminding us of [13, Remark 2, p. 6], Alex for pointing out the improved attack in Section 5.3, and Phil for encouraging us to consider the case of small e more closely. A.O. was supported in part by Alexandra Boldyreva's NSF CAREER award 0545659 and NSF Cyber Trust award 0831184 and thanks her for her support. A.S. was supported in part by NSF awards #0747294, 0729171.

References

[1] Abdalla M., Bellare M., Rogaway P.: The Oracle Diffie-Hellman Assumptions and an Analysis of DHIES. In CT-RSA 2001.

[2] Barak B., Shaltiel R., Tromer E.: True Random Number Generators Secure in a Changing Environment. In CHES 2003.

[3] Bellare M., Rompel J.: Randomness-Efficient Oblivious Sampling. In: FOCS 1994. ACM (1994)
[4] Bellare M., Rogaway P.: Random oracles are practical: A paradigm for designing efficient protocols. the In Conference on Computer and Communications Security. ACM (1993)
[5] Bellare M., Rogaway P.: Optimal asymmetric encryption: How to encrypt with RSA. In: EUROCRYPT 1994. LNCS. Springer (1994)
[6] Boldyreva A., Fehr S., O'Neill A.: On notions of security for deterministic encryption, and efficient constructions without random oracles. In: CRYPTO 2008. LNCS. Springer (2008)
[7] Boldyreva A., Fischlin M.: Analysis of random oracle instantiation scenarios for OAEP and other practical schemes. In: CRYPTO 2005. LNCS. Springer (2005)
[8] Boldyreva A., Fischlin M.: On the security of OAEP. In: ASIACRYPT 2006. LNCS. Springer (2006)
[9] Boneh D.: Simplified OAEP for the RSA and Rabin functions. In: CRYPTO 2001. LNCS. Springer (2001)
[10] Brown D.: What hashes make RSA-OAEP secure? In Cryptology ePrint Archive, Report 2006/223 (2006)
[11] Boldyreva A., Cash C., Fischlin M., Warinschi B.: Efficient private bidding and auctions with an oblivious third party. In: ASIACRYPT 2009.
[12] Cachin, C.: Efficient private bidding and auctions with an oblivious third party. In: CCS 1999. ACM (1999)
[13] Cachin, C., Micali, S., Stadler, M.: Computationally private information retrieval with polylogarithmic communication. In: EUROCRYPT 1999. Full verion at http://www.zurich.ibm.com/ cca/papers/cpir.pdf
[14] Canetti R.: Towards realizing random oracles: Hash functions that hide all partial information. In: CRYPTO 1997. LNCS. Springer (1997)
[15] Canetti R., Dakdouk R.: Extractable Perfectly One-Way Functions. In: ICALP 2008.
[16] Canetti R., Goldreich O., Halevi S.: The random oracle methodology, revisited. J. ACM 51(4): 557-594 (2004)
[17] Canetti R., Micciancio D., and Reingold O.: Perfectly one-way probabilistic hash functions. In: STOC 1998. ACM (1998)
[18] Coppersmith D.: Small solutions to polynomial equations, and low exponent RSA vulnerabilities. In J. Cryptology. Vol. 10. Springer (1997)
[19] Coron J-S., Joye M., Naccache D., Paillier P.: New Attacks on PKCS #1 v1.5 Encryption. In *EUROCRYPT 2000*. LNCS. Springer (2000)
[20] Coron J-S., Joye M., Naccache D., Paillier P.: Universal Padding Schemes for RSA, In *CRYPTO 2002*. LNCS. Springer (2002)
[21] Dodis Y., Oliveira R., Pietrzak K.: On the Generic Insecurity of the Full Domain Hash. In *CRYPTO 2005*. LNCS. Springer (2005)
[22] Dodis Y., Smith A: Correcting errors without leaking partial information. In: STOC 2005. ACM (2005)

[23] Fujisaki E., Okamoto T., Pointcheval D., Stern J.: RSA-OAEP is secure under the RSA assumption. In: J. Cryptology 17(2): 81-104 (2004)
[24] Gentry, C., Mackenzie, P., Ramzan, Z.: Password authenticated key exchange using hidden smooth subgroups. In: CCS 2005. ACM (2005)
[25] Hemenway B., Ostrovsky R.: Public-key locally-decodable codes. In *CRYPTO 2008*. LNCS. Springer (2008)
[26] Kazukuni K., Imai H.: OAEP++: A Very Simple Way to Apply OAEP to Deterministic OW-CPA Primitives. In Cryptology ePrint Archive, Report 2002/130 (2002)
[27] Kiltz E., O'Neill A., Smith A.: Instantiability of RSA-OAEP under Chosen-Plaintexts Attacks. Full version of this paper.
[28] Kiltz E., Pietrzak K.: The Group of Signed Quadratic Residues and Applications. In: CRYPTO 2009. LNCS. Springer (2009)
[29] Kiltz E., Pietrzak K.: On the security of padding-based encryption schemes (or: Why we cannot prove OAEP secure in the standard model). In: EUROCRYPT 2009. LNCS. Springer (2009)
[30] Kiltz E., Pietrzak K.: Personal Communication, 2009.
[31] Lenstra A. K.: Unbelievable security : Matching AES security using public key systems. In: *ASIACRYPT 2001* LNCS Springer (2001)
[32] May A.: Using LLL-Reduction for Solving RSA and Factorization Problems: A Survey. LLL+25 Conference in honour of the 25th birthday of the LLL algorithm, 2007.
[33] Herrmann M., May A.: Solving Linear Equations Modulo Divisors: On Factoring Given Any Bits. In ASIACRYPT 2008. LNCS. Springer (2008)
[34] Paillier P., Villar J.: Trading one-wayness against chosen-ciphertext security in factoring-based encryption. In ASIACRYPT 2006. LNCS. Springer (2006)
[35] Pandey O., Pass R., Vaikuntanathan V.: Adaptive One-Way Functions and Applications In CRYPTO 2008. LNCS. Springer (2008)
[36] Peikert C., Waters B.: Lossy trapdoor functions and their applications. In: *STOC 2008*. ACM (2008)
[37] RSA Laboratories Public-Key Cryptography Standards. `www.rsa.com/rsalabs/pkcs/`
[38] Rivest R., Shamir A., Adelman L.: A method for obtaining public-key cryptosystems and digital signatures. Technical Report MIT/LCS/TM-82 (1977)
[39] Rivest R., Shamir A., Adelman L.: Cryptographic communications system and method. U.S. Patent 4,405,829 (1983)
[40] Schridde C,. Freisleben B.: On the validity of the Φ-Hiding Assumption in cryptographic protocols. In: ASIACRYPT 2008. LNCS. Springer (2008)
[41] Shoup V.: OAEP Reconsidered. In: *J. Cryptology* 15(4): 223-249 (2002)
[42] Trevisan L., Vadhan S.: Extracting Randomness from Samplable Distributions. In: FOCS 2000. ACM (2000)
[43] Yilek S., Rescorla E., Shacham H., Enright B., Savage S.: When Private Keys are Public: Results from the 2008 Debian OpenSSL Debacle. In: IMC 2009.