Toward RSA-OAEP without Random Oracles

Nairen Cao[1] Adam O'Neill[2] Mohammad Zaheri[3]

February 11, 2020

In Memoriam: John C. O'Neill (1953–2018).

Abstract

We show new partial and full instantiation results *under chosen-ciphertext security* for the widely implemented and standardized RSA-OAEP encryption scheme of Bellare and Rogaway (EUROCRYPT 1994) and two variants. Prior work on such instantiations either showed negative results or settled for "passive" security notions like IND-CPA. More precisely, recall that RSA-OAEP adds redundancy and randomness to a message before composing two rounds of an underlying Feistel transform, whose round functions are modeled as random oracles (ROs), with RSA. Our main results are:

- Either of the two oracles (while still modeling the other as a RO) can be instantiated in RSA-OAEP under IND-CCA2 using mild standard-model assumptions on the round functions and generalizations of algebraic properties of RSA shown by Barthe, Pointcheval, and Báguelin (CCS 2012). The algebraic properties are only shown to hold at practical parameters for small encryption exponent ($e = 3$), but we argue they have value for larger e as well.
- Both oracles can be instantiated simultaneously for two variants of RSA-OAEP, called "t-clear" and "s-clear" RSA-OAEP. For this we use extractability-style assumptions in the sense of Canetti and Dakdouk (TCC 2010) on the round functions, as well as novel yet plausible "XOR-type" assumptions on RSA. While admittedly strong, such assumptions may nevertheless be necessary at this point to make positive progress.

In particular, our full instantiations evade impossibility results of Shoup (J. Cryptology 2002), Kiltz and Pietrzak (EUROCRYPT 2009), and Bitansky *et al.* (STOC 2014). Moreover, our results for s-clear RSA-OAEP yield the most efficient RSA-based encryption scheme proven IND-CCA2 in the standard model (using bold assumptions on cryptographic hashing) to date.

[1] Dept. of Computer Science, Georgetown University, Email: nc645@georgetown.edu
[2] Dept. of Computer Science, Georgetown University, Email: adam@cs.georgetown.edu
[3] Dept. of Computer Science, Georgetown University, Email: mz394@georgetown.edu

Contents

1 Introduction

In this paper, we show new partial and full instantiations *under chosen-ciphertext attack* (CCA) for the RSA-OAEP encryption scheme [12] and some variants. This helps explain why the scheme, which so far has only been shown to have such security in the random oracle (RO) model, has stood up to cryptanalysis despite the existence of "uninstantiable" RO model schemes and other negative results. It also leads to the fastest CCA-secure RSA-based public-key encryption scheme in the standard model (where one assumes standard-model properties of cryptographic hash functions) to date. We now discuss some background and motivation before an overview of our results.

1.1 Background and Motivation

In the random oracle (RO) model of Bellare and Rogaway [11], every algorithm has oracle access to the same truly random functions. This model has been enormously enabling in the design of practical protocols for various goals; examples include public-key encryption [11, 12, 51], digital signatures [11, 13], and identity-based encryption [26]. When a RO model scheme is implemented, one "instantiates" the oracles, that is, replaces their invocations with invocations of functions with publicly-available code. Thus, there are many possible "instantiations" of a protocol, depending on the choice of the latter. To obtain a practical instantiation, it was suggested by [11] to build these functions from cryptographic hashing in an appropriate way. We call this the *canonical instantiation*. The *RO model thesis* of [11] is that if a protocol is secure in the RO model then its canonical instantiation remains secure in the standard (RO devoid) sense.

Unfortunately, the RO model thesis has been refuted in a strong sense, starting with the work of Canetti *et al.* [33]. These works show that there exist RO model schemes for which *any* instantiation, let alone the canonical one, yields a scheme that can be broken efficiently in the standard model. However, the consensus of the community is that such schemes always seem contrived or artificial in some way. Indeed, RO model schemes that have been standardized have stood up to decades of cryptanalysis. If the RO model thesis is false, what explains this? This leads to what may be called the *practical RO model thesis:* For a "practical" scheme proven secure in the RO model scheme, its canonical instantiation remains secure in the standard model. However, from a scientific standpoint this thesis is unsatisfactory because it lacks a *definition* of "practical."[1] This shortcoming is the starting point for our work.

1.2 Our Thesis

Candidate differentiating properties. It seems problematic to try to define practicality in the above sense. Instead, we propose some candidate properties that we conjecture to differentiate schemes to which the RO model thesis applies from those to which it does not. Here are some such properties, some of which are inspired by our work described below:

1. There exist standard-model properties of the constituent functions that together suffice to prove security of the scheme, ideally as well as realizations of such functions under standard assumptions.
2. *Each* individual constituent function can be separately instantiated as above, while possibly modeling the others as ROs.
3. Variants of the scheme that fall under the same framework satisfy one of the above properties.
4. There exist constructions of standard-model hash functions that allow to prove security of the scheme when replacing the ROs, ideally these constructions being under standard assumptions.

The revised thesis. Our *revised RO model thesis* is that a scheme satisfying one of the above properties is such that the canonical instantiation yields a secure scheme in the standard model, where we relax the notion of instantiation to allow stronger assumptions on non-RO constituent functions. That is, "constituent functions" refers not only to those modeled as ROs but possibly other functions associated with the scheme, like RSA. Thus, one may search for novel assumptions on RSA, for example. Indeed, if one looks at the question of why some RSA-based RO scheme is secure in practice, it could very well have to do with properties of RSA (which has a lot of algebraic structure) beyond mere one-wayness. We have seen the same strategy used to explain security of

[1] Here we do not mean "practical" in the sense of efficient enough to use in practice, but rather "does not do anything contrived."

schemes, without transitioning between the RO and standard models, for example with Chaum's blind signature scheme [9] and Damgård's ElGamal [39]. It was also advocated by Pandey *et al.* [63] to resolve some long-standing theoretical questions.

It is also worth mentioning that there are impossibility results in the standard model for RSA-OAEP [58] and RSA-FDH, RSA-PSS [41, 40]. However, these are *black-box* impossibility results that demonstrate that a proof treating the functions as black-boxes cannot suffice. As in other areas of cryptography [2] this motivates looking at non-blackbox assumptions.

1.3 Discussion of The Properties and Our Goals

Our focus: RSA-OAEP. We focus our study on whether the RO model thesis applies to a very influential scheme, namely *RSA-OAEP* [12]. Roughly, RSA-OAEP is defined as follows. RSA-OAEP encrypts a message as $f(s\|t)$ where f is the RSA function, where for functions $\mathcal{G}$ and $\mathcal{H}$ (originally modeled as ROs) we have $s = \mathcal{G}(r)\oplus m\|0^\zeta$ for randomness $r \in \{0,1\}^\rho$ and message $m \in \{0,1\}^\mu$, $t = \mathcal{H}(s)\oplus r$. (We denote $s = s_1\|s_2$.) Thus, we would like to examine whether RSA-OAEP satisfies the properties listed above.

The first property. Here we seek standard model properties of RSA, $\mathcal{G}$, and $\mathcal{H}$ that suffice to prove IND-CCA. For this property, we mentioned that ideally we would also have theoretical realizations of such functions under standard assumptions. We make it clear that we do not advocate *using* these theoretical realizations in practice, but they would show that the goal is not impossible to achieve. The importance of this is illustrated by the fact that the most general forms of assumptions such as correlation intractability (CI) [33] and universal computational extraction (UCE) [7, 28] have been shown (likely) impossible. (But special cases of CI and UCE which suffice for the schemes considered remain plausible [7, 28, 30].) Unfortunately, we do not know how to achieve the first property for RSA-OAEP, even without such theoretical realizations.

The second property. The second property asks for so-called "partial instantiations" for each one of $\mathcal{G}$ or $\mathcal{H}$, while still modeling the other as a RO. Partial instantiations are valuable because ROs are used in different ways in a scheme, and instantiating one of them isolates a property it relies on. Moreover, we ask that *every* oracle can be (separately) instantiated. This has provable implications in practice as well, as now an attacker would need to exploit weakness in the *interaction* between these functions in order to break the scheme in the standard model. In our eyes this makes a standard model attack much less plausible. We show that RSA-OAEP satisfies this property under suitable assumptions.

The third property. The third property is more subjective than the others, as it hinges on what constitutes a scheme falling under the same framework. The aim is to capture the scheme designers' intent or their general approach. Again, the idea is not to use the modified schemes in practice necessarily (although one certainly could if the efficiency penalty is acceptable), but to validate the framework more than simply proving the original scheme is secure in the RO model. An upshot is that this approach can indeed lead to variants of the scheme that offer better security with similar efficiency. We show the third property holds for RSA-OAEP, and in fact our results for one of our variants, namely s-clear RSA-OAEP, leads to the most efficient IND-CCA secure scheme in the standard model, albeit under bold assumptions on cryptographic hashing.

The fourth property. Note that this property differs from the first in that it does not require giving higher-level properties that the hash functions should satisfy in order to make the scheme secure. Thus, it does not really give insight into what properties hash functions used in the canonical instantiation should satisfy to do this. Still, existence of such hash functions refutes uninstantiability of the scheme, showing that the job of the hash functions in making the scheme secure is at least plausible. As with the first property, we leave it as an open problem to show this for RSA-OAEP. We note that this property has been shown for other RO model schemes in, *e.g*, [54, 72].

We proceed to describe our approach and results in more detail.

1.4 Using PA + IND-CPA

Using PA + IND-CPA. A common thread running through our analyses is the use of *plaintext awareness* (PA) [12, 5, 10]. PA captures the intuition that an adversary who produces a ciphertext must "know" the corresponding plaintext. It is not itself a notion of privacy, but, at a high level, combined with IND-CPA it implies IND-CCA. We use this approach to obtain modularity in proofs, isolate assumptions needed, and make

overall analyses more tractable. Moreover, while it seems that PA necessitates using knowledge assumptions, this is somewhat inherent anyway due to black-box impossibility results discussed below.

FLAVORS AND IMPLICATIONS. PA comes in various flavors: PA-RO [5], and PA0, PA1, and PA2 [10]. PA-RO refers to a notion in the RO model, while PA0, PA1, and PA2 refer to standard model notions that differ in what extent the adversary can query its decryption or encryption oracles. (In particular, in PA2 the adversary can query for encryptions of unknown plaintexts.) Similarly, IND-CCA comes in flavors [66, 5]: IND-CCA0, IND-CCA1, and IND-CCA2. We use that [5, 10] show that IND-CPA + PA-RO implies IND-CCA2 in the RO model, IND-CPA + PA0 implies IND-CCA1 with one decryption query, IND-CPA + PA1 implies IND-CCA1, and IND-CPA + PA2 implies IND-CCA2.

1.5 Partial Instantiation Results

HIGH-LEVEL APPROACH. We first give partial instantiation results of RSA-OAEP under IND-CCA2. Such results have been sought after in prior work [29, 21, 22] but have proven negative results or settled for weaker security notions. The heroes for us here are new generalizations of the notions of "second-input extractability" (SIE) and "common-input extractability" (CIE) proven by Barthe *et al.* [3] to hold for small-exponent RSA ($e = 3$). SIE says that an RSA image point can be inverted given a sufficiently-long (depending on e) part of the preimage, whereas CIE says that two RSA images can be inverted if the preimages share a common part. They were used by [3] where the "part" is the least-significant bits to analyze a no-redundancy, one-round version of RSA-OAEP in the RO model. The assumptions are proven via Coppersmith's algorithm to find small roots of a univariate polynomial modulo N [35].

We show that generalized versions where the "part" refers to some of the middle or most-significant bits, rather than least-significant bits, is useful for analyzing RSA-OAEP more generally. We show these versions also hold for small-exponent RSA, but based on the *bivariate* Coppersmith algorithm [35, 19, 37]. Moreover, despite the similarity of assumptions, our proof strategies in the partial instantiations are somewhat different than that of Barthe *et al.* [3]. Another interesting point is that while (generalized) SIE and CIE hold for $e = 3$, we argue they have practical value for larger e as well. Namely, while $e > 3$ would require an impractical "part" length using Coppersmith's technique, they could possibly hold for practical parameters via other (in particular, non-blackbox) techniques. At least, we do not see how to refute that, which could lend insight into why there is no IND-CCA2 attack on the scheme for general e. [2]

RESULTS AND INTUITION. Namely, we show partial instantiations of both oracles $\mathcal{G}, \mathcal{H}$ under very mild assumptions on the round functions — roughly, that $\mathcal{G}$ is a pseudorandom generator and $\mathcal{H}$ is a hardcore function for RSA, respectively — in both cases assuming RSA is SIE and CIE. We first prove IND-CPA security in these cases. Interestingly, the instantiation of $\mathcal{G}$ under IND-CPA uses that RSA is SIE while the instantiation of $\mathcal{H}$ does not, the intuition being that in the latter case we assume $\mathcal{H}$ is a hardcore function so its output masks $r \in \{0,1\}^\rho$ used in the challenge ciphertext unconditionally. Now for PA-RO, in both cases we use SIE and CIE, but wrt. different bits of the input. In the case of instantiating $\mathcal{G}$, it is wrt. the redundancy bits s_2. Intuitively, for a decryption query there are two cases. Firstly, that it has a *different* r-part than the challenge and therefore this must have been queried to the RO, in which case the SIE extractor works. Secondly, that it has the *same* r-part as the challenge, but it therefore shares s_2, in which case the CIE extractor works. In the case of instantiating $\mathcal{H}$, there are again two cases for an encryption query depending on whether it shares the same s-part of the challenge or not; thus the assumption is wrt. the whole s-part.

1.6 Full Instantiation Results

HIGH-LEVEL APPROACH. We next give full instantiation results for two variants of RSA-OAEP, called t-clear and s-clear RSA-OAEP. Prior results on t-clear RSA-OAEP [22] showed only partial instantiations or relatively weak security notions, and s-clear RSA-OAEP was only considered indirectly by Shoup [70] for negative results. In t-clear RSA-OAEP, a message is encrypted as $f(s_1)\|s_2\|t$ where f is the RSA function $s_1\|s_2 = \mathcal{G}(r)\oplus m\|0^\varsigma$ for randomness $r \in \{0,1\}^\rho$ and message $m \in \{0,1\}^\mu$, $t = \mathcal{H}(s_1\|s_2)\oplus r$. Here we divide s into $s_1\|s_2$, where $s_2 \in \{0,1\}^\varsigma$, so the name "$t$-clear" while consistent with prior work [22], is somewhat of a misnomer. On the other hand, in s-clear RSA OAEP a message is encrypted as $s\|f(t)$. One of the heroes for us here is a hierarchy of "extractability"

[2] Moreover, we conjecture this is different from the case of "lossiness" [64, 57] as shown for RSA and used to analyze IND-CPA security of RSA-OAEP in [57]. Namely, to get sufficient lossiness it seems to inherently require large e, since the *only* way to make RSA parameters lossy is to have $e \mid \phi(N)$.

notions we define and assume for the round functions, called EXT-RO, EXT0, EXT1, EXT2, roughly paralleling PA-RO, PA0, PA1, PA2 respectively, and significantly generalizing prior work [31, 32]. Besides this parallel, our generalizations consider adversaries that output only part of an image point or an image point along with part of a pre-image. These are bold assumptions to make on (functions constructed out of) cryptographic hash functions, but, as discussed above, we believe studying their implications is justified. In the case of s-clear, another hero is a family of new"XOR-type" assumptions we introduce, and give intuitive justifications for in light of the multiplicative structure of RSA. Again, we view part of our contribution as putting forth novel assumptions that the research community can analyze (say in the generic ring model) in the future.

We make several remarks about our results, particularly how they avoid known impossibility results, before detailing them:

- Extractability is a non-blackbox assumption (saying for every adversary there exists a non-blackbox "extractor") so we avoid the impossibility result of Kiltz and Pietrzak [58].[3]. That is, the fact we use extractable hash functions (extractability being an intuitive property used in the original RO model proof) is somewhat unavoidable.

- While extractability of $\mathcal{H}$ would *prima facie* be false, we use it only in a plausible way for a cryptographic hash function. Namely, the adversary also outputs *part of the preimage*. Extractability assumptions we use on $\mathcal{G}$, even where the adversary outputs only part of an image point, remain plausible as it is an expanding function with a sparse range (usually constructed something like $\mathcal{G}(x) = (\mathcal{H}(0\|x)\|\mathcal{H}(1\|x), \ldots)$.

- For extractability we use only bounded key-independent auxiliary input (basically, the keys for the other functions in the scheme), so we avoid the impossibility result of Bitansky *et al.* [17]. Moreover, the key-dependent auxiliary information is just one image query (at least in the proof of IND-CCA2).

- Our "XOR-type" assumptions on RSA avoid a negative result of Shoup [70], showing that there is an attack if the general trapdoor permutation is "XOR-malleable."[4]

- We typically use the various forms of extractability in combination with (at least) collision-resistance, so that the extractor returns the "right" preimage. The collision-resistant construction of [61] based on knowledge assumptions, albeit where the adversary outputs the entire image point, is on the lowest level of our hierarchy (EXT0); furthermore, it is not known to work when the adversary outputs part of the image point. Any theoretical constructions for higher levels (EXT1, EXT2) are similarly open. We hope these are targeted in future work.

Results and intuition for t-clear. Our results for t-clear RSA-OAEP are weaker than those for s-clear RSA-OAEP. First, for t-clear we prove IND-CPA for high-entropy, public key independent messages, under mild assumptions on the round functions, namely that $\mathcal{H}$ is a hardcore function for RSA and $\mathcal{G}$ is a pseudorandom generator. Intuitively, the high-entropy requirement comes from the fact that the adversary attacking $\mathcal{H}$ needs to know r to prepare its challenge ciphertext, so the randomness of the input to $\mathcal{H}$ needs to come from m. (We could avoid it using the stronger assumption of UCE as per the result of [7], which could be viewed as a hedge.) Furthermore, m needs to be public-key independent so as to not bias the output. Then we can prove PA0 based on forms of EXT0 for $\mathcal{G}$ and $\mathcal{H}$, the intuition being that the plaintext extractor first extracts from the part $\mathcal{G}(r)$ that is left in clear by the redundancy to get r and then runs the extractor for $\mathcal{H}$ on $t \oplus r$ from which it can compute m, with the above part of the pre-image to get s. Note that when running the extractor here and below we have to be careful that the constructed extractor uses the same coins as the starting one for consistency (otherwise we won't end up with the right extractor). We can also prove PA1, although we have to make an extractability assumption directly on the padding scheme.[5] Interestingly, even this approach does not work for PA2, which we leave completely open for t-clear (cf. Remark 7.6).

Results and intuition for s-clear. We find s-clear is much more friendly to a full instantiation by making novel but plausible assumptions on RSA. One is XOR-nonmalleability (XOR-NM), saying that from $\mathcal{F}(x)$ it is hard to find some $\mathcal{F}(x')$ and z such that $z = x \oplus x'$. Another is XOR-indistinguishability (XOR-IND), saying for

[3] As acknowledged by the authors there was a bug in the proceedings version of this paper, but this has been fixed for the full version [59].

[4] In more detail, note that for s-clear the "overall" TDP (including the part output in the clear) is not partial one-way [46] so their security proof does *not* apply. In fact, Shoup [70] considers the scheme in his proof that RSA-OAEP is not IND-CCA2-secure for general one-way TDPs, exhibiting the above-mentioned attack.

[5] At a very high level, we can prove EXT0 of $\mathcal{G}, \mathcal{H}$ implies EXT0 for the padding scheme, but we do not know how to do this for EXT1 because of an "extractor blow-up" problem.

Scheme	Assumptions on OAEP	Assumptions on $\mathcal{F}$	Security	Size	Ref
RSA-OAEP	**$\mathcal{G}$: PRG and $\mathcal{H}$: RO**	**OW, SIE and CIE**	**IND-CCA2**	n	**Section 3**
RSA-OAEP	**$\mathcal{G}$: RO and $\mathcal{H}$: PHCF**	**OW, SIE and CIE**	**IND-CCA2**	n	**Section 3**
RSA-OAEP	$\mathcal{G}$: t-wise independent	Lossy TDP	IND-CPA	n	[57]
RSA-OAEP	$\mathcal{G}, \mathcal{H}$: UCE	OW	IND-CPA-KI	n	[7]
RSA-OAEP *t*-**clear**	**$\mathcal{G}$: PRG, EXT0 and NCR $\mathcal{H}$: HCF, EXT0 and CR**	**OW**	**$IND-CCA0-KI**	$3n + 3k$	**Full version**
RSA-OAEP *t*-**clear**	**OAEP : EXT1 and NCR $\mathcal{G}$: PRG and $\mathcal{H}$: HCF**	**OW**	**$IND-CCA1-KI**	$3n + 3k$	**Full version**
RSA-OAEP *t*-clear	$\mathcal{G}$: PRG and NCR $\mathcal{H}$: RO	OW	IND-CCA2	$n + k$	[22]
RSA-OAEP *t*-clear	$\mathcal{G}$: RO $\mathcal{H}$: NM PRG with hint	OW	IND-CCA2	$n + k$	[22]
RSA-OAEP *t*-clear	$\mathcal{G}$: PRG and NCR $\mathcal{H}$: NM PRG with hint	OW	$NM-CPA	$n + k$	[22]
RSA-OAEP *s*-**clear**	**$\mathcal{G}$: PRG, EXT1 and NCR**	**XOR-IND0**	**IND-CCA1**	$2n + k + \mu$	**Section 7**
RSA-OAEP *s*-**clear**	**$\mathcal{G}$: PRG, EXT2 and NCR $\mathcal{H}$: CR**	**XOR-IND1,2 and XOR-NM0**	**IND-CCA2**	$2n + k + \mu$	**Section 7**

Figure 1: Instantiability results for RSA-OAEP, where n is modulus length, k is security param and μ is message length. Typically $n = 2048, k = 128$ and $\mu = 128$.

random x and adversarially-chosen z one cannot tell $\mathcal{F}(x)$ from $\mathcal{F}(x \oplus z)$ given "hint" $\mathcal{G}(x)$. In our results, $\mathcal{G}$ is a PRG, which we show also implies $\mathcal{G}$ is a HCF for $\mathcal{F}$. So, the notion can be viewed as an extension of the classical notion of HCF. In fact, we use XOR-IND just to show IND-CPA. The intuition is that it allows breaking the dependency of s in the input to OAEP with the input to RSA. The proofs of PA0 and PA1 are very similar, and showcase one reason s-clear is much more friendly to a full instantiation, namely it heavily depends on the extractability of $\mathcal{G}$. That is, if $\mathcal{G}$ is suitably extractable, the plaintext extractor can simply recover r and then compute the plaintext as $s \oplus \mathcal{G}(r)$. For PA2, one has to be careful as when the adversary makes an encryption query, the plaintext extractor should call the image oracle for $\mathcal{G}$, where in addition to $\mathcal{G}(x)$ for random x it receives the hint of RSA on x. We show that if RSA is XOR-IND then this implies the adversary can get the whole ciphertext as a hint to simulate the encryption oracle. Then we also have the worry about the adversary querying "mauled" ciphertexts to the extract oracle. Intuitively, if the r-part is the same then it cannot run the extractor for $\mathcal{G}$, but we show this violates XOR-NM of RSA. On the other hand, if the s-part is the same then we cannot break XOR-NM but this creates a collision for $\mathcal{G}$.

1.7 Discussion and Perspective

We summarize and compare our results to prior work in Figure 1. Note that we get a lot of mileage from assuming the trapdoor permutation is specifically RSA, whereas prior work, which has mostly shown negative results CCA-style security notions, went for a general approach. We also highlight that while our assumptions on both RSA and the round functions for our full instantiability results are expectedly stronger than what we need for partial instantiations, they still compare favorably to prior work. In particular, while our assumption of EXT2 for $\mathcal{G}$ in our s-clear result is already "PA2-flavored," prior work [22, Definition 3.3] made CCA-style assumptions on the round functions even to obtain relatively weak notions of non-malleability. It can also be viewed as a strengthening of "adaptive" (CCA-style) security notions on one-way functions [63, 56].[6] Plus, it is not clear how to get an IND-CCA2 encryption scheme from EXT2 functions in a simpler way.

1.8 Related Work

RO MODEL RESULTS. Results about security of $\mathcal{F}$-OAEP for an abstract TDP $\mathcal{F}$ with applications to RSA-OAEP in the RO model were shown in [12, 70, 46]. Ultimately, these works showed RSA-OAEP is IND-CCA2 secure in the RO model assuming only one-wayness of RSA, but with a loose security reduction. Interestingly, Shoup [70] considers s-clear RSA-OAEP indirectly in a negative result about RSA-OAEP with a general one-way TDP. Security of t-clear RSA-OAEP (under the name "RSA-OAEP++") has been analyzed in the RO model by Boldyreva, Imai and Kobara [23], who show tight security in the multi-challenge setting.

[6] These works do not precisely match our setting as [63] consider keyless functions and [56] consider functions with a trapdoor.

Partial instantiation results. Canetti [29] conjectured that his notion of perfect one-wayness sufficed to instantiate *one* of the two oracles in $\mathcal{F}$-OAEP. This was disproved in general by Boldyreva and Fischlin [21], but their results do not contradict ours because they use a contrived TDP $\mathcal{F}$. Subsequently, Boldyreva and Fischlin [22] gave partial instantiations for t-clear $\mathcal{F}$-OAEP under stronger assumptions on the round functions.

Full instantiation results. Brown [27] and Paillier and Villar [62] showed negative results for proving RSA-OAEP is IND-CCA secure in restricted models, and Kiltz and Pietrzak [58] showed a general black-box impossibility result. As mentioned above, their results do not contradict ours because we use non-blackbox assumptions. Moving to weaker notions, Kiltz *et al.* [56] show IND-CPA security of RSA-OAEP using lossiness [64], while Bellare, Hoang, and Keelveedhi [7] show RSA-OAEP is IND-CPA secure for public-key independent messages assuming the round functions meet their notion of universal computational extraction. Boldyreva and Fischlin [22] show a weak form of non-malleability for t-clear $\mathcal{F}$-OAEP, again using very strong assumptions on the round functions. Lewko *et al.* [60] show IND-CPA security of the RSA PKCS v1.5 scheme, with the bounds later being corrected and improved by Smith and Zhang [71].

Candidate instantiability assumptions. General notions for function families geared towards instantiating ROs that have been proposed include correlation intractability [33, 30], extractable hash functions [31, 32, 15, 17], perfect one-wayness [29, 34, 44], seed incompressibility [50], non-malleability [20, 1], and universal computational extraction (UCE) [7, 28, 8].

1.9 Organization

In Section 2, we give the preliminaries. In Section 3, we formalize the algebraic properties of RSA we use and our partial instantiation results for RSA-OAEP. In Section 4, we give a new hierarchy of extractable functions. In Section 5, we abstract out some properties of the OAEP padding scheme we use. Then, in Section 7 we give novel "XOR-type" assumptions on RSA and combine them with the above to give our full instantiation result s-clear RSA-OAEP. Due to space constraints, our results for t-clear RSA-OAEP are deferred to the supplementary materials. We also defer all detailed proofs to the supplementary materials.

2 Preliminaries and Some Generalizations

We overview notations and definitions we use that are mostly from prior work.

2.1 Notation and Conventions

For a probabilistic algorithm A, by $y \leftarrow\!\!\$\ A(x)$ we mean that A is executed on input x and the output is assigned to y. We sometimes use $y \leftarrow A(x;r)$ to make A's random coins explicit. We denote by $\Pr\left[A(x) = y : x \leftarrow\!\!\$\ X\right]$ the probability that A outputs y on input x when x is sampled according to X. We denote by $[A(x)]$ the set of possible outputs of A when run on input x. The security parameter is denoted $k \in \mathbb{N}$. Unless otherwise specified, all algorithms must run in probabilistic polynomial-time (PPT) in k, and an algorithm's running-time includes that of any overlying experiment as well as the size of its code. Integer parameters often implicitly depend on k. The length of a string s is denoted $|s|$. We denote by $s|_i^j$ the i-th least significant bits (LSB) to j-th least significant bits of s (including the i-th and j-th bits), where $1 \le i \le j \le |s|$. For convenience, we denote by $s|_\ell = s|_1^\ell$ the ℓ least significant bits of s and $s|^\ell = s|_{|s|-\ell}^{|s|}$ the ℓ most significant bits (MSB) of s, for $1 \le \ell \le |s|$. We write P_X for the distribution of random variable X and $P_X(x)$ for the probability that X puts on value x, i.e. $P_X(x) = \Pr[X = x]$. We denote by U_ℓ the uniform distribution on $\{0,1\}^\ell$. We write U_S for the uniform distribution on the set S. Vectors are denoted in boldface, for example $\mathbf{x}$. If $\mathbf{x}$ is a vector then $|\mathbf{x}|$ denotes the number of components of $\mathbf{x}$ and $\mathbf{x}[i]$ denotes its i-th component, for $1 \le i \le |\mathbf{x}|$. For convenience, we extend algorithmic notation to operate on each vector of inputs component-wise. For example, if A is an algorithm and $\mathbf{x}, \mathbf{y}$ are vectors then $\mathbf{z} \leftarrow\!\!\$\ A(\mathbf{x}, \mathbf{y})$ denotes that $\mathbf{z}[i] \leftarrow\!\!\$\ A(\mathbf{x}[i], \mathbf{y}[i])$ for all $1 \le i \le |\mathbf{x}|$. Let X be a random variable taking values on a common finite domain. The *min-entropy* of a random variable X is $\mathrm{H}_\infty(X) = -\log(\max_x \Pr[X = x])$.

2.2 Public-Key Encryption and its Security

Public-key encryption. A *public-key encryption scheme* PKE with message space Msg is a tuple of algorithms $(\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$. The key-generation algorithm Kg on input 1^k outputs a public key pk and matching secret key sk. The encryption algorithm Enc on inputs pk and a message $m \in \mathsf{Msg}(1^k)$ outputs a ciphertext c. The

Game $\text{IND-ATK}^{A}_{\mathsf{PKE}}(k)$
$b \leftarrow\!\!\$\ \{0,1\}\ ;\ (pk, sk) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathcal{O}_1(\cdot)}(1^k, pk)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b(1^k, pk)$
$c \leftarrow\!\!\$\ \mathsf{Enc}(pk, m_b)$
$d \leftarrow\!\!\$\ A_2^{\mathcal{O}_2(\cdot)}(pk, c, state)$
Return $(b = d)$

Figure 2: **Game to define** IND-ATK **security.**

Game $\text{PA-RO}^{A,\mathsf{Ext}}_{\mathsf{PKE}}(k)$	**Procedure** $\textsc{Enc}(pk, \mathcal{M})$
$b \leftarrow\!\!\$\ \{0,1\}\ ;\ i \leftarrow 1\ ;\ j \leftarrow 1$	$m \leftarrow\!\!\$\ \mathcal{M}(1^k, pk)$
$(pk, sk) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	$c \leftarrow\!\!\$\ \mathsf{Enc}^{\text{RO}(\cdot,2)}(pk, m)$
$b' \leftarrow\!\!\$\ A^{\text{RO}(\cdot,1),\textsc{Enc}(pk,\cdot),\mathcal{D}(sk,\cdot)}(pk)$	$\mathbf{c}[i] \leftarrow c\ ;\ i \leftarrow i+1$
Return $(b = b')$	Return c
Procedure $\text{RO}(x, i)$	**Procedure** $\mathcal{D}(sk, c)$
If $H[x] = \perp$ then $H[x] \leftarrow\!\!\$\ \{0,1\}^\ell$	If $c \in \mathbf{c}$ then return $\perp$
If $i = 1$ then	$m_0 \leftarrow \mathsf{Dec}(sk, c)$
$\mathbf{x}[j] \leftarrow x\ ;\ \mathbf{h}[j] \leftarrow H[x]\ ;\ j \leftarrow j+1$	$m_1 \leftarrow\!\!\$\ \mathsf{Ext}^{\text{RO}(\cdot,3)}(\mathbf{x}, \mathbf{h}, \mathbf{c}, c, pk)$
Return $H[x]$	Return m_b

Figure 3: **Game to define PA-RO security.**

deterministic decryption algorithm Dec on inputs sk and ciphertext c outputs a message m or $\perp$. We require that for all $(pk, sk) \in [\mathsf{Kg}(1^k)]$ and all $m \in \mathsf{Msg}(1^k)$, $\mathsf{Dec}(sk, (\mathsf{Enc}(pk, m)) = m$ with probability 1.

Security of public-key encryption [48, 67]. Let $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a public key encryption scheme and $A = (A_1, A_2)$ be an adversary. Let $\mathcal{M}$ be a PPT algorithm that takes inputs 1^k and a public key pk to return a message $m \in \mathsf{Msg}(1^k)$. For ATK $\in$ {CPA, CCA1, CCA2 } we associate the experiment in Figure 2 for every $k \in \mathbb{N}$. Define the *ind-atk advantage* of A against PKE as

$$\mathbf{Adv}^{\text{ind-atk}}_{\mathsf{PKE},A}(k) = 2 \cdot \Pr\left[\text{IND-ATK}^{A}_{\mathsf{PKE}}(k) \Rightarrow 1\right] - 1\ .$$

If atk = cpa, then $\mathcal{O}_1(\cdot) = \varepsilon$, and $\mathcal{O}_2(\cdot) = \varepsilon$. We say PKE is *secure under chosen-plaintext attack* (IND-CPA) if $\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{PKE},A}(k)$ is negligible in k for all PPT A.
Similarly, if atk = cca1, then $\mathcal{O}_1(\cdot) = \mathsf{Dec}(sk, \cdot)$, and $\mathcal{O}_2(\cdot) = \varepsilon$; if atk = cca2, then $\mathcal{O}_1(\cdot) = \mathsf{Dec}(sk, \cdot)$, and $\mathcal{O}_2(\cdot) = \mathsf{Dec}(sk, \cdot)$. In the case of cca2, A_2 is not allowed to ask $\mathcal{O}_2$ to decrypt c. We say that PKE is secure under non-adaptive chosen-ciphertext attack or IND-CCA1 (resp. adaptive chosen-ciphertext attack or IND-CCA2), if $\mathbf{Adv}^{\text{ind-cca1}}_{\mathsf{PKE},A}(k)$ (resp. $\mathbf{Adv}^{\text{ind-cca2}}_{\mathsf{PKE},A}(k)$) is negligible in k for all PPT A.

PA-RO security. We first define plaintext-awareness in the RO model following [6], which builds on the definition in [12]. Note that PA-RO combined with IND-CPA security is strictly stronger than IND-CCA2 security in general. Let $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a public key encryption scheme and let $\mathcal{M}$ be a PPT algorithm that takes as inputs 1^k and a public key pk, and outputs a message $m \in \mathsf{Msg}(1^k)$. To adversary A and extractor Ext, we associate the experiment in Figure 3 for every $k \in \mathbb{N}$. We say that PKE is PA-RO secure if for every PPT adversary A there exists an extractor Ext such that

$$\mathbf{Adv}^{\text{pa-ro}}_{\mathsf{PKE},A,\mathsf{Ext}}(k) = 2 \cdot \Pr\left[\text{PA-RO}^{A,\mathsf{Ext}}_{\mathsf{PKE}}(k) \Rightarrow 1\right] - 1\ .$$

is negligible in k.

Remark 2.1 Our definition of plaintext awareness in the random oracle model differs from the definition given in [6] in the following way. In our definition, we are giving the extractor access to the random oracle. We observe that the analogous result of [6, Theorem 4.2] that IND-CPA and PA-RO together imply IND-CCA2 still holds for our modified definition, since in the proof the IND-CPA adversary could query its own random oracle to answer to the random oracle queries of the extractor.

We now turn to definitions of plaintext awareness in the standard model, following [10].

Game $\mathrm{PA1}^{A,\mathsf{Ext}}_{\mathsf{PKE}}(k)$	**Procedure** $\mathcal{D}(sk, c)$
$b \leftarrow\!\!\$\ \{0,1\}$	$m_0 \leftarrow \mathsf{Dec}(sk, c)$
$(pk, sk) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	$(m_1, state) \leftarrow\!\!\$\ \mathsf{Ext}(state, c)$
$r \leftarrow\!\!\$\ \mathsf{Coins}(1^k)$	Return m_b
$state \leftarrow (pk, r)$	
$b' \leftarrow A^{\mathcal{D}(sk,\cdot)}(pk; r)$	
Return $(b = b')$	

Figure 4: **Games to define PA1 security.**

Game $\mathrm{PA2}^{A,\mathsf{Ext}}_{\mathsf{PKE}}(k)$	**Procedure** $\mathcal{D}(sk, c)$
$(pk, sk) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	If $c \in \mathbf{c}$ then return $\perp$
$b \leftarrow\!\!\$\ \{0,1\}\ ;\ i \leftarrow 1$	$m_0 \leftarrow \mathsf{Dec}(sk, c)$
$r \leftarrow\!\!\$\ \mathsf{Coins}(k)$	$(m_1, state) \leftarrow\!\!\$\ \mathsf{Ext}(state, \mathbf{c}, c)$
$state \leftarrow (pk, r)$	Return m_b
$b' \leftarrow A^{\mathcal{D}(sk,\cdot),\textsc{Enc}(pk,\cdot)}(pk; r)$	**Procedure** $\textsc{Enc}(pk, \mathcal{M})$
Return $(b = b')$	$m \leftarrow\!\!\$\ \mathcal{M}(1^k, pk)$
	$c \leftarrow\!\!\$\ \mathsf{Enc}(pk, m)$
	$\mathbf{c}[i] \leftarrow c\ ;\ i \leftarrow i + 1$
	Return c

Figure 5: **Games to define PA2 security.**

PA1 SECURITY. Let $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a public key encryption scheme. To adversary A and extractor Ext, we associate the experiment in Figure 4 for every $k \in \mathbb{N}$. We say that PKE is PA1 secure if for every PPT adversary A with coin space Coins there exists an extractor Ext such that,

$$\mathbf{Adv}^{\mathrm{pa1}}_{\mathsf{PKE},A,\mathsf{Ext}}(k) = 2 \cdot \Pr\left[\mathrm{PA1}^{A,\mathsf{Ext}}_{\mathsf{PKE}}(k) \Rightarrow 1\right] - 1\ .$$

is negligible in k.

PA0 SECURITY. We define PA0 similarly to PA1, except A is only allowed to make a single oracle query. Let PA0 be the corresponding experiment, and define

$$\mathbf{Adv}^{\mathrm{pa0}}_{\mathsf{PKE},A,\mathsf{Ext}}(k) = 2 \cdot \Pr\left[\mathrm{PA0}^{A,\mathsf{Ext}}_{\mathsf{PKE}}(k) \Rightarrow 1\right] - 1\ .$$

We say PKE is PA0 secure if for every PPT adversary A there exists an extractor Ext such that $\mathbf{Adv}^{\mathrm{pa0}}_{\mathsf{PKE},A,\mathsf{Ext}}(k)$ is negligible in k.

PA2 SECURITY. Let $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a public-key encryption scheme. To adversary A and extractor Ext, we associate the experiment in Figure 5 for every $k \in \mathbb{N}$. We say that PKE is PA2 secure if for every PPT adversary A there exists an extractor Ext such that,

$$\mathbf{Adv}^{\mathrm{pa2}}_{\mathsf{PKE},A,\mathsf{Ext}}(k) = 2 \cdot \Pr\left[\mathrm{PA2}^{A,\mathsf{Ext}}_{\mathsf{PKE}}(k) \Rightarrow 1\right] - 1\ .$$

is negligible in k.

Remark 2.2 Our PA2 definition comes from [10]. Other than PA1 adversary, we will give PA2 adversary extra access to encryption oracle. This models the ability IND-CCA2 adversary obtains ciphertext without knowing the randomness.

2.3 Trapdoor Permutations and Their Security

TRAPDOOR PERMUTATIONS. A trapdoor permutation family with domain TDom is a tuple of algorithms $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ that work as follows. Algorithm Kg on input a unary encoding of the security parameter 1^k outputs a pair (f, f^{-1}), where $f\colon \mathsf{TDom}(k) \to \mathsf{TDom}(k)$. Algorithm Eval on inputs a function f and $x \in \mathsf{TDom}(k)$ outputs

$y \in \mathsf{TDom}(k)$. We often write $f(x)$ instead of $\mathsf{Eval}(f, x)$. Algorithm Inv on inputs a function f^{-1} and $y \in \mathsf{TDom}(k)$ outputs $x \in \mathsf{TDom}(k)$. We often write $f^{-1}(y)$ instead of $\mathsf{Inv}(f^{-1}, y)$. We require that for any $(f, f^{-1}) \in [\mathsf{Kg}(1^k)]$ and any $x \in \mathsf{TDom}(k)$, $f^{-1}(f(x)) = x$. We call $\mathcal{F}$ an n-bit trapdoor permutation family if $\mathsf{TDom} = \{0,1\}^n$. We will think of the RSA trapdoor permutation family [68] n-bit for simplicity, although its domain is $\mathbb{Z}_N^*$ for an n-bit integer N. Additionally, for convenience we define the following. For an ν-bit trapdoor permutation family $\mathcal{F}$ and $\ell \in \mathbb{N}$, we define $\mathcal{F}|_\ell = (\mathsf{Kg}|_\ell, \mathsf{Eval}|_\ell, \mathsf{Inv}|_\ell)$ as the $(\nu+\ell)$-bit trapdoor permutation families such that for all $k \in \mathbb{N}$, all $(f|_\ell, f^{-1}|_\ell) \in [\mathsf{Kg}|_\ell(1^k)]$, and all $x \in \{0,1\}^{\nu+\ell}$, we have $f|_\ell(x) = f(x|^{n-\ell})\|x|_\ell$, and analogously for $\mathcal{F}|^\ell$.

One-wayness. Let $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a trapdoor permutation family with domain TDom. We say $\mathcal{F}$ is *one-way* if for every PPT inverter I:

$$\mathbf{Adv}^{\mathrm{owf}}_{\mathcal{F},I}(k) = \Pr_{\substack{(f,f^{-1}) \xleftarrow{\$} \mathsf{Kg}(1^k) \\ x \xleftarrow{\$} \mathsf{TDom}(k)}} \left[\begin{array}{c} x' \leftarrow I(f, f(x)) \\ x' = x \end{array} \right] .$$

is negligible in k.

Partial one-wayness. Let $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a trapdoor permutation family with domain TDom. We say $\mathcal{F}$ is ζ-*partial one way* if for every PPT inverter I:

$$\mathbf{Adv}^{\mathrm{pow}}_{\mathcal{F},I}(k) = \Pr_{\substack{(f,f^{-1}) \xleftarrow{\$} \mathsf{Kg}(1^k) \\ x \xleftarrow{\$} \mathsf{TDom}(k)}} \left[\begin{array}{c} x' \leftarrow I(f, f(x)) \\ x' = x|^\zeta \end{array} \right] .$$

is negligible in k. It is shown in [45] that for RSA one-wayness implies partial one-wayness but the reduction is lossy.

2.4 Function Families and Associated Security Notions

Function families. A function family with domain $\mathsf{F.Dom}$ and range $\mathsf{F.Rng}$ is a tuple of algorithms $\mathcal{F} = (\mathcal{K}_F, F)$ that work as follows. Algorithm $\mathcal{K}_F$ on input a unary encoding of the security parameter 1^k outputs a key K_F. Deterministic algorithm F on inputs K_F and $x \in \mathsf{F.Dom}(k)$ outputs $y \in \mathsf{F.Rng}(k)$. We alternatively write $\mathcal{F}$ as a function $\mathcal{F}\colon \mathcal{K}_F \times \mathsf{F.Dom} \to \mathsf{F.Rng}$. We call $\mathcal{F}$ an ℓ-*injective* function if for all distinct $x_1, x_2 \in \mathsf{F.Dom}(k)$ and $K_F \in [\mathcal{K}_F(1^k)]$, we have $F(K_F, x_1)|_\ell \neq F(K_F, x_2)|_\ell$.

Collision resistance. Let $\mathcal{H}\colon \mathcal{K}_H \times \mathsf{HDom} \to \mathsf{HRng}$ be a function family. We say $\mathcal{H}$ is *collision resistant* (CR) if for any PPT adversary A:

$$\mathbf{Adv}^{\mathrm{cr}}_{\mathcal{H},A}(k) = \Pr_{K_H \xleftarrow{\$} \mathcal{K}_H(1^k)} \left[\begin{array}{c} (x_1, x_2) \leftarrow A(K_H) \\ x_1, x_2 \in \mathsf{HDom}(k) \end{array} \wedge \begin{array}{c} \mathcal{H}(K_H, x_1) = \mathcal{H}(K_H, x_2) \\ x_1 \neq x_2 \end{array} \right] .$$

is negligible in k. Again, this is a standard notion that can be realized in a variety of ways, in particular it is one of the basic properties believed for cryptographic hash function.

Near-collision resistance. Let $\mathcal{H}\colon \mathcal{K}_H \times \mathsf{HDom} \to \mathsf{HRng}$ be a function family. For $\ell \in \mathbb{N}$, we say $\mathcal{H}$ is *near-collision resistant* with respect to ℓ-least significant bits of the outputs (NCR_ℓ) if for any PPT adversary A:

$$\mathbf{Adv}^{\text{n-cr}_\ell}_{\mathcal{H},A}(k) = \Pr_{K_H \xleftarrow{\$} \mathcal{K}_H(1^k)} \left[\begin{array}{c} (x_1, x_2) \leftarrow A(K_H) \\ x_1, x_2 \in \mathsf{HDom}(k) \end{array} \wedge \begin{array}{c} \mathcal{H}(K_H, x_1)|_\ell = \mathcal{H}(K_H, x_2)|_\ell \\ x_1 \neq x_2 \end{array} \right] .$$

is negligible in k. We note that our definition differs slightly from [22] as both x_1, x_2 are adversarially chosen. In terms of feasibility, the same construction based on one-way permutations given in [22] works in our case as well. Similarly, we define NCR^ℓ where the adversary try to find collision on the ℓ-most significant bits of the output.

Pseudorandom generators. Let $\mathcal{G} : \mathcal{K}_G \times \mathsf{GDom} \to \mathsf{GRng}$ be a function family. To adversary A, we associate the experiment in Figure 6 for every $k \in \mathbb{N}$. We say that $\mathcal{G}$ is a *pseudorandom generator* if for every PPT adversary A,

$$\mathbf{Adv}^{\mathrm{prg}}_{\mathcal{G},A}(k) = 2 \cdot \Pr\left[\,\text{PRG-DIST}^A_{\mathcal{G}}(k) \Rightarrow 1\,\right] - 1 \ .$$

is negligible in k. This is a standard notion in theory and can be heuristically constructed from a cryptographic hash function in straightforward ways.

Game $\text{PRG-DIST}^A_{\mathcal{G}}(k)$
$b \xleftarrow{\$} \{0,1\}$
$K_G \xleftarrow{\$} \mathcal{K}_G(1^k)$; $x \xleftarrow{\$} \mathsf{GDom}(k)$
$r_0 \leftarrow G(K_G, x)$; $r_1 \xleftarrow{\$} \mathsf{GRng}(k)$
$b' \xleftarrow{\$} A(K_G, r_b)$
Return $(b = b')$

Figure 6: **Games to define PRG-DIST security.**

Game $\text{HCF-DIST}^{A,\mathbf{X}}_{\mathcal{F},\mathcal{H}}(k)$
$b \xleftarrow{\$} \{0,1\}$
$K_H \xleftarrow{\$} \mathcal{K}_H(1^k)$; $(f, f^{-1}) \xleftarrow{\$} \mathsf{Kg}(1^k)$
$(\mathbf{x}, \alpha) \xleftarrow{\$} \mathbf{X}(k)$; $\mathbf{h}_0 \leftarrow H(K_H, \mathbf{x})$
$\mathbf{h}_1 \xleftarrow{\$} (\mathsf{HRng}(k))^{\times|\mathbf{x}|}$
$b' \xleftarrow{\$} A(K_H, f, f(\mathbf{x}), \alpha, \mathbf{h}_b)$
Return $(b = b')$

Figure 7: **Games to define HCF-DIST security.**

Pseudorandom generators with image verifier. Let $\mathcal{G} : \mathcal{K}_G \times \mathsf{GDom} \to \mathsf{GRng}$ be a function family. *Pseudorandom generators with ℓ-bit image verifier* is similar to *pseudorandom generator* except we will give adversary A with oracle access to $\mathcal{V}_\ell$, where $\mathcal{V}_\ell$ is an ℓ-bit image verifier that on input y works as follows:

$$\mathcal{V}_\ell(y) = \begin{cases} 1 & \text{if } \exists x \colon y = G(K_G, x)|_\ell \\ 0 & \text{otherwise} \end{cases} .$$

Note that adversary A is not allowed to query for the challenge to the image verifier oracle. We say that $\mathcal{G}$ is a pseudorandom generator with ℓ-bit image verifier (VPRG_ℓ) if for every PPT adversary A,

$$\mathbf{Adv}^{\text{vprg}_\ell}_{\mathcal{G},A}(k) = 2 \cdot \Pr\left[\,\text{VPRG-DIST}^A_{\mathcal{G}}(k) \Rightarrow 1\,\right] - 1 \ .$$

is negligible in k. In our results we do not require $\mathcal{V}_\ell$ to be efficient, so they are (we believe) plausible for constructions based on cryptographic hash functions. It is weaker than the "adaptivity" assumption made in [63].

Hardcore functions. We define a notion of hardcore functions for non-uniform, correlated distributions as in [47], but we extend it to consider auxiliary input as well. Let $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a one-way trapdoor permutation family with domain TDom. Let $\mathcal{H} \colon \mathcal{K}_H \times \mathsf{TDom} \to \mathsf{HRng}$ be a function family. For $k \in \mathbb{N}$, let $\mathbf{X}(k)$ be a distribution on input vector in $\mathsf{TDom}(k)$ and auxiliary information $\alpha \in \{0,1\}^*$. To attacker A and distribution $\mathbf{X}(k)$, we associate the experiment in Figure 7 for every $k \in \mathbb{N}$. We say that $\mathcal{H}$ is a *hardcore function* for the trapdoor permutation family $\mathcal{F}$ on a family of such distributions $\mathbb{X}$ if for every $\mathbf{X}(k) \in \mathbb{X}(k)$ and for every PPT adversary A,

$$\mathbf{Adv}^{\text{hcf}}_{\mathcal{F},\mathcal{H},\mathbf{X},A}(k) = 2 \cdot \Pr\left[\,\text{HCF-DIST}^{A,\mathbf{X}}_{\mathcal{F},\mathcal{H}}(k) \Rightarrow 1\,\right] - 1 \ .$$

is negligible in k. For messages drawn from a block-source, if $\mathcal{F}$ is sufficiently lossy in the sense of [65] then a universal hash function meets this notion. Additionally, a $2t$-wise independent function meets this notion for t arbitrarily correlated, high-entropy messages if $\mathcal{F}$ loses a $1 - o(1)$ fraction of its input. It is an open problem to construct such a hardcore function for an unbounded number of arbitrarily correlated, high-entropy messages. However, we see it as plausible that a cryptographic hash function meets this definition.

Partial hardcore functions. For convenience, we also generalize the notion of hardcore function in the following way. Let $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be n-bit trapdoor permutation family. Let $\mathcal{H} \colon \mathcal{K}_H \times \{0,1\}^{n-\ell} \to \mathsf{HRng}$ be a function family, for some $\ell < n$. To attacker A, we associate the experiment in Figure 8 for every $k \in \mathbb{N}$. We say that $\mathcal{H}$ is a *ℓ-partial hardcore function* for the trapdoor permutation family $\mathcal{F}$ if for every PPT adversary A,

$$\mathbf{Adv}^{\text{phcf}}_{\mathcal{F},\mathcal{H},A}(k) = 2 \cdot \Pr\left[\,\text{PHCF-DIST}^A_{\mathcal{F},\mathcal{H}}(k) \Rightarrow 1\,\right] - 1 \ .$$

is negligible in k. Note if $(f(x), x|_{n-\ell})$ is a one-way function of x, then $\mathcal{H}$ is a ℓ-partial hardcore function for $\mathcal{F}$ when $\mathcal{H}$ is a computational randomness extractor [38]. This is plausible for the case that $\mathcal{F}$ is RSA when $n - \ell$

Game PHCF-DIST$^{A}_{\mathcal{F},\mathcal{H}}(k)$
$b \leftarrow\!\!\$\ \{0,1\}$
$K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)\ ;\ (f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$
$x \leftarrow\!\!\$\ \{0,1\}^n\ ;\ h_0 \leftarrow H(K_H, x\|^{\ell})$
$h_1 \leftarrow\!\!\$\ \mathsf{HRng}(k)$
$b' \leftarrow\!\!\$\ A(K_H, f, f(x), x\|_{n-\ell}, h_b)$
Return $(b = b')$

Figure 8: **Games to define PHCF-DIST security.**

$\underline{\mathsf{Kg}(1^k)}$	$\underline{\mathsf{Enc}(pk, m\|\|r)}$	$\underline{\mathsf{Dec}(sk, c)}$
$(\pi, \hat{\pi}) \leftarrow\!\!\$\ \Pi$	$(\pi, f) \leftarrow pk$	$(\hat{\pi}, f^{-1}) \leftarrow pk$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	$y \leftarrow\!\!\$\ \pi(m\|\|r)$	$y \leftarrow f^{-1}(c)$
$pk \leftarrow (\pi, f)$	$c \leftarrow f(y)$	$m \leftarrow \hat{\pi}(y)$
$sk \leftarrow (\hat{\pi}, f^{-1})$	Return c	Return m
Return (pk, sk)		

Figure 9: **Padding based encryption scheme** $\mathsf{PAD}[\mathcal{F}] = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$**.**

is small enough that Coppersmith's techniques do not apply. This means $n - \ell \leq n(e-1)/e - \log 1/\epsilon$ such that $N^{\epsilon} \geq 2^k$ for security parameter k.

2.5 The OAEP Framework

Padding scheme. We define a general notion of padding scheme following [12, 58]. For $\nu, \rho, \mu \in \mathbb{N}$, the associated *padding scheme* is a triple of deterministic algorithms $\mathsf{PAD} = (\Pi, \mathsf{PAD}, \mathsf{PAD}^{-1})$ defined as follows. Algorithm Π on input a unary encoding of the security parameter 1^k outputs a pair $(\pi, \hat{\pi})$ where $\pi : \{0,1\}^{\mu+\rho} \to \{0,1\}^{\nu}$ and $\hat{\pi} : \{0,1\}^{\nu} \to \{0,1\}^{\mu} \cup \{\perp\}$ such that π is injective and for all $m \in \{0,1\}^{\mu}$ and $r \in \{0,1\}^{\rho}$ we have $\hat{\pi}(\pi(m\|r)) = m$. Algorithm PAD on inputs π and $m \in \{0,1\}^{\mu}$ outputs $y \in \{0,1\}^{\nu}$. Algorithm PAD^{-1} on inputs a mapping $\hat{\pi}$ and $y \in \{0,1\}^{\nu}$ outputs $m \in \{0,1\}^{\mu}$ or $\perp$.

Padding-based encryption. Let PAD be a padding transform from domain $\{0,1\}^{\mu+\rho}$ to range $\{0,1\}^{\nu}$. Let $\mathcal{F}$ be a TDP with domain $\{0,1\}^{\nu}$. The associated *padding-based encryption scheme* is a triple of algorithms $\mathsf{PAD}[\mathcal{F}] = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ defined in Figure 9.

OAEP padding scheme. We recall the OAEP padding scheme [12]. Let message length μ, randomness length ρ, and redundancy length ζ be integer parameters, and $\nu = \mu + \rho + \zeta$. Let $\mathcal{G}\colon \mathcal{K}_G \times \{0,1\}^{\rho} \to \{0,1\}^{\mu+\zeta}$ and $\mathcal{H}\colon \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^{\rho}$ be function families. The associated *OAEP padding scheme* is a triple of algorithms $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}] = (\mathcal{K}_{\mathsf{OAEP}}, \mathsf{OAEP}, \mathsf{OAEP}^{-1})$ defined as follows. On input 1^k, $\mathcal{K}_{\mathsf{OAEP}}$ returns (K_G, K_H) where $K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$ and $K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$, and $\mathsf{OAEP}, \mathsf{OAEP}^{-1}$ are as defined in Figure 10.

OAEP encryption scheme and variants. Slightly abusing notation, we denote by $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ the OAEP-based encryption scheme $\mathcal{F}$-OAEP with $n = \nu$. We also consider two other OAEP-based encryption schemes, called *t-clear* and *s-clear* $\mathcal{F}$-OAEP, and denoted $\mathsf{OAEP}_{\mathsf{t\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|_{\zeta+\rho}]$ and $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|^{\mu+\zeta}]$. Here $n = \mu$ and $n = \rho$, respectively. We often write $\mathsf{OAEP}_{\mathsf{t\text{-}clear}}$ and $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}$ instead of $\mathsf{OAEP}_{\mathsf{t\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|_{\zeta+\rho}]$ and $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|^{\mu+\zeta}]$. We typically think of $\mathcal{F}$ as RSA, and all our results apply to this case under suitable assumptions. Note that, following prior work, despite its name t-clear $\mathcal{F}$-OAEP we actually apply $\mathcal{F}$ to only the μ most significant bits of the output of the underlying padding scheme, leaving the redundancy part of s in the clear as well.

3 Partial Instantiation Results for RSA-OAEP

In this section, we give partial instantiations of either $\mathcal{G}$ or $\mathcal{H}$ for RSA-OAEP under IND-CCA2. Our results use only mild standard model properties of $\mathcal{G}$ or $\mathcal{H}$. They also use (generalizations of) algebraic properties of RSA proven by Barthe *et al.* [3] for small enough e. For example, using a 2048-bit modulus and encrypting a 128-bit AES key, our results hold for $e = 3$. They may be true for larger e; at least, we do not know how they can be disproved. Note that our results first necessitate a separate proof of IND-CPA — the standard model IND-CPA

Algorithm $\mathsf{OAEP}_{(K_G,K_H)}(m\|r)$	**Algorithm** $\mathsf{OAEP}^{-1}_{(K_G,K_H)}(x)$
$s \leftarrow (m\|0^\zeta) \oplus G(K_G, r)$	$s\|t \leftarrow x$
$t \leftarrow r \oplus H(K_H, s)$	$r \leftarrow t \oplus H(K_H, s)$
$x \leftarrow s\|t$	$m' \leftarrow s \oplus G(K_G, r)$
Return x	If $m'|_\zeta = 0^\zeta$ return $m'|^\mu$
	Else return $\perp$

Figure 10: **OAEP padding scheme** $\mathsf{OAEP}[G, H]$**.**

results of Kiltz *et al.* [57] and Bellare *et al.* [7] are not suitable, the first requiring large e and the second holding only for public-key independent messages.

3.1 Algebraic Properties of RSA

We first give the (generalizations of) algebraic properties of RSA from Barthe *et al.* [3] that we use and their parameters. Note that they used these assumptions to analyze security of a zero-redudancy one-round version of RSA-OAEP. We show these are useful for analyzing security of RSA-OAEP more generally.

Second-input extractability. Let $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a trapdoor permutation family with domain $\{0,1\}^n$. For $1 \le i \le j \le n$, we say $\mathcal{F}$ is *(blackbox)* (i,j)*-second-input-extractable* (BB (i,j)-SIE) if there exists an efficient extractor $\mathcal{E}$ such that for every $k \in \mathbb{N}$, every $f \in [\mathsf{Kg}(1^k)]$, and every $x \in \{0,1\}^n$, extractor $\mathcal{E}$ on inputs $f, f(x), x|_{i+1}^j$ outputs x. We often write ζ-SIE instead of $(n-\zeta, n)$-SIE.

Common-input extractability. Let $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a trapdoor permutation family with domain $\{0,1\}^n$. For $1 \le i \le j \le n$, we say $\mathcal{F}$ is *(blackbox)* (i,j)*-common-input-extractable* if there exists an efficient extractor $\mathcal{E}$ such that for every $k \in \mathbb{N}$, every $f \in [\mathsf{Kg}(1^k)]$, and every $x_1, x_2 \in \mathsf{TDom}(k)$, extractor $\mathcal{E}$ on inputs $f, f(x_1), f(x_2)$ outputs (x_1, x_2) if $x_1|_{i+1}^j = x_2|_{i+1}^j$. We often write ζ-CIE instead of $(n-\zeta, n)$-CIE.

Comparison to Barthe *et al.* Compared to [3], we generalize the notions of SIE and CIE to consider arbitrary runs of consecutive bits. That is, [3] only considers the most significant bits; *i.e.*, ζ-SIE and ζ-CIE in our notation. We also explicitly call the notions *blackbox* to emphasize the extractor does not make use of the code or random coins of an adversary producing its input. Interestingly, we define analogous notions in Section 4 where this is not the case.

Parameters. Barthe *et al.* [3] show via the Coppersmith algorithm [35] that RSA is ζ-SIE and ζ-CIE for sufficiently large ζ. Specifically, they show RSA is ζ_1-SIE for $\zeta_1 > n(e-1)/e$, and ζ_2-CIE for $\zeta_2 > n(e^2-1)/e^2$. We show that a generalization to runs of arbitrary consecutive bits holds in Appendix A. Specifically, in Appendix A we show that RSA is (i,j)-SIE for $(j-i) > n(e-1)/e$, and (i,j)-CIE for $(j-i) > n(e^2-1)/e^2$. In our partial instantiation results for RSA-OAEP, $j-i$ refers to the length of the redundancy ζ.

3.2 Main Results

Main results. We now give our main results, namely partial instantiations for RSA-OAEP of either oracle $\mathcal{G}$ or $\mathcal{H}$. These results refer to IND-CCA security for simplicity, whereas we actually prove PA-RO + IND-CPA.

Theorem 3.1 Let n, μ, ζ, ρ be integer parameters. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ be a pseudorandom generator and $\mathcal{H} : \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be a RO. Let $\mathcal{F}$ be a family of trapdoor permutations with domain $\{0,1\}^n$, where $n = \mu + \zeta + \rho$. Suppose $\mathcal{F}$ is one-way, $(\mu+\zeta)$-second input and $(\mu+\zeta)$-common input extractable. Then $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ is IND-CCA2 secure. In particular, for any adversary A, there is an adversary D and an inverter I such that

$$\mathbf{Adv}^{\text{ind-cca2}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A}(k) \le 2 \cdot \mathbf{Adv}^{\text{owf}}_{\mathcal{F},I}(k) + 10 \cdot \mathbf{Adv}^{\text{prg}}_{\mathcal{G},D}(k) + \frac{2p}{2^{\mu+\zeta}} + \frac{4q}{2^\zeta} \ .$$

where q is the total number of the decryption queries and p is the total number of RO queries made by A.

Theorem 3.2 Let n, μ, ζ, ρ be integer parameters. Let $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be a hash function family and $\mathcal{G} : \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ be a RO. Let $\mathcal{F}$ be a family of trapdoor permutations with domain $\{0,1\}^n$, where $n = \mu + \zeta + \rho$. Suppose $\mathcal{F}$ is $(\rho, \rho+\zeta)$-second input and $(\rho, \rho+\zeta)$-common input extractable. Suppose further $\mathcal{H}$ is a $(\mu+\zeta)$-partial hardcore function for $\mathcal{F}$. Then $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ is IND-CCA2. In particular, for any adversary

Games $G_1(k)$, $G_2(k)$	**Games** $G_3(k)$, $G_4(k)$
$b \leftarrow\!\!\$\ \{0,1\}$; $K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$	$b \leftarrow\!\!\$\ \{0,1\}$; $K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_G, f)$	$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_G, f)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{RO}_1(\cdot)}(1^k, pk)$	$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{RO}_1(\cdot)}(1^k, pk)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{RO}_1(\cdot)}(1^k, pk)$	$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{RO}_1(\cdot)}(1^k, pk)$
$r \leftarrow\!\!\$\ \{0,1\}^\rho$; $x \leftarrow G(K_G, r)$	$r \leftarrow\!\!\$\ \{0,1\}^\rho$; $x \leftarrow G(K_G, r)$
$s \leftarrow x \oplus (m_b \| 0^\zeta)$	$s \leftarrow x \oplus (m_b \| 0^\zeta)$; $t \leftarrow\!\!\$\ \{0,1\}^\rho$
If $H[s] \neq \perp$ then	$z \leftarrow t \oplus r$; $H[s] \leftarrow z$; $c \leftarrow f(s \| t)$
$\quad \mathsf{bad}_1 \leftarrow \mathsf{true}$; $H[s] \leftarrow\!\!\$\ \{0,1\}^\rho$	$d \leftarrow\!\!\$\ A_2^{\mathrm{RO}_2(\cdot)}(c, state)$
Else $H[s] \leftarrow\!\!\$\ \{0,1\}^\rho$	Return $(b = d)$
$z \leftarrow H[s]$; $t \leftarrow z \oplus r$; $c \leftarrow f(s \| t)$	**Procedure** $\mathrm{RO}_1(v)$
$d \leftarrow\!\!\$\ A_2^{\mathrm{RO}_2(\cdot)}(c, state)$	Return $\mathrm{RO}(v)$
Return $(b = d)$	**Procedure** $\mathrm{RO}_2(v)$
Procedure $\mathrm{RO}_1(v)$	If $v = s$ then
Return $\mathrm{RO}(v)$	$\quad \mathsf{bad}_2 \leftarrow \mathsf{true}$; return $\overline{\mathrm{RO}}(v)$
Procedure $\mathrm{RO}_2(v)$	Return $\mathrm{RO}(v)$
Return $\mathrm{RO}(v)$	

Figure 11: **Games G_1–G_4 in the proof of Theorem 3.3.**

$A = (A_1, A_2)$, there exists an adversary B such that

$$\mathbf{Adv}^{\text{ind-cca2}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A}(k) \leq 2 \cdot \mathbf{Adv}^{\text{phcf}}_{\mathcal{F},\mathcal{H},B}(k) + \frac{2p}{2^\rho} + \frac{4q}{2^\zeta} \ .$$

where q the total number of the decryption queries and p is the total number of RO queries made by A.

The proofs of both theorems follow from below.

Parameters for RSA-OAEP. We discuss when our results support RSA-OAEP encryption of an AES key of appropriate length, based on Subsection 3.1. The main requirement is encryption exponent $e = 3$. In this case, with length 2048 bits we can use randomness and message length 128 bits, and for modulus length 4096 we can use randomness length 256. The choice that $e = 3$ is sometimes used in practice but it is an interesting open problem to extend our results to other common choices such as $e = 2^{16} + 1$. In particular, it is a reasonable conjecture that results for SIE and CIE hold in this case for the same parameters.

3.3 Partial Instantiation of G

We first show how to instantiate $\mathcal{G}$ when modeling $\mathcal{H}$ as a RO. In particular, we show $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ is IND-CPA + PA-RO when $\mathcal{G}$ is a pseudorandom generator and $\mathcal{F}$ is one-way, (blackbox) $(\mu + \zeta)$-SIE and $(\mu + \zeta)$-CIE.

IND-CPA result. Under IND-CPA, we show a tight reduction when $\mathcal{G}$ is a pseudorandom generator and $\mathcal{F}$ is one-way and $(\mu + \zeta)$-SIE. Alternatively, in Appendix B we give result where $\mathcal{F}$ is only partial one-way, but the reduction is lossy. Notes that it is shown in [45] that one-wayness of RSA implies partial one-wayness, but the reduction is even more lossy, while SIE and CIE unconditionally hold for appropriate parameters.

Theorem 3.3 Let n, μ, ζ, ρ be integer parameters. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ be a pseudorandom generator and $\mathcal{H} : \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be a RO. Let $\mathcal{F}$ be a family of trapdoor permutations with domain $\{0,1\}^n$, where $n = \mu + \zeta + \rho$. Suppose $\mathcal{F}$ is one-way and $(\mu + \zeta)$-second input extractable. Then $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ is IND-CPA. In particular, for any adversary $A = (A_1, A_2)$, there are an adversary D and an inverter I such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A}(k) \leq 2 \cdot \mathbf{Adv}^{\text{owf}}_{\mathcal{F},I}(k) + 6 \cdot \mathbf{Adv}^{\text{prg}}_{\mathcal{G},D}(k) + \frac{2q}{2^{\mu+\zeta}} \ .$$

where q is the total number of RO queries made by A. Furthermore, the running time of D and I are about that of A plus the time to run SIE extractor.

Games $G_5(k)$	**Games** $G_6(k)$
$b \leftarrow\!\!\$\ \{0,1\}$; $K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$	$b \leftarrow\!\!\$\ \{0,1\}$; $K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_G, f)$	$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_G, f)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{RO}_1(\cdot)}(1^k, pk)$	$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{RO}_1(\cdot)}(1^k, pk)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{RO}_1(\cdot)}(1^k, pk)$	$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{RO}_1(\cdot)}(1^k, pk)$
$r \leftarrow\!\!\$\ \{0,1\}^\rho$; $x \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$	$r \leftarrow\!\!\$\ \{0,1\}^\rho$; $s \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$
$s \leftarrow x \oplus (m_b \| 0^\zeta)$	$x \leftarrow s \oplus (m_b \| 0^\zeta)$
$t \leftarrow\!\!\$\ \{0,1\}^\rho$; $c \leftarrow f(s\|t)$	$t \leftarrow\!\!\$\ \{0,1\}^\rho$; $c \leftarrow f(s\|t)$
$d \leftarrow\!\!\$\ A_2^{\mathrm{RO}_2(\cdot)}(c, state)$	$d \leftarrow\!\!\$\ A_2^{\mathrm{RO}_2(\cdot)}(c, state)$
Return $(b = d)$	Return $(b = d)$
Procedure $\mathrm{RO}_1(v)$	**Procedure** $\mathrm{RO}_1(v)$
Return $\mathrm{RO}(v)$	Return $\mathrm{RO}(v)$
Procedure $\mathrm{RO}_2(v)$	**Procedure** $\mathrm{RO}_2(v)$
If $v = s$ then return $\overline{\mathrm{RO}}(v)$	If $v = s$ then return $\overline{\mathrm{RO}}(v)$
Return $\mathrm{RO}(v)$	Return $\mathrm{RO}(v)$

Figure 12: **Games G_5, G_6 in the proof of Theorem 3.3.**

Algorithm $B(K_G, x)$	**Procedure** $\mathrm{ROSIM}_1(v)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $\mathsf{out} \leftarrow 0$	If $H[v] = \perp$ then
$pk \leftarrow (K_G, f)$; $b \leftarrow\!\!\$\ \{0,1\}$	$\quad H[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{ROSIM}_1(\cdot)}(1^k, pk)$	Return $H[v]$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{ROSIM}_1(\cdot)}(1^k, pk)$	
$s \leftarrow x \oplus (m_b \| 0^\zeta)$	
If $H[s] \neq \perp$ then $\mathsf{out} \leftarrow 1$	
Return out	

Figure 13: **Adversary B in the proof of Theorem 3.3.**

Proof: Consider games G_1–G_6 in Figures 11–12. Each game maintains two independent random oracles RO and $\overline{\mathrm{RO}}$. Procedure RO maintains a local array H as follows:

Procedure $\mathrm{RO}(v)$
If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
Return $H[v]$

For simplicity, we omit the code of $\mathrm{RO}, \overline{\mathrm{RO}}$ in the games. In each game, we use RO_1 to denote the oracle interface of adversary A_1 and message samplers $\mathcal{M}_0, \mathcal{M}_1$, and we use RO_2 to denote the oracle interface of adversary A_2. Game G_1 corresponds to game IND-CPA$_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}]}$. Then

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A}(k) \leq 2 \cdot \Pr[G_1(k) \Rightarrow 1] - 1 \ .$$

We now explain the game chain. Game G_2 is identical to game G_1, except in the encryption of message m_b. Namely, if either adversary A_1 or message sampler $\mathcal{M}_b$ queries s to their random oracle RO_1, it chooses a fresh random value for $H[s]$. Games G_1 and G_2 are identical-until-bad_1, and thus from the Fundamental Lemma of Game-Playing [14],

$$\Pr\left[\, G_1(k) \Rightarrow 1 \,\right] - \Pr\left[\, G_2(k) \Rightarrow 1 \,\right] \leq \Pr\left[\, G_2(k) \text{ sets } \mathsf{bad}_1 \,\right] \ .$$

Now, consider adversary B attacking the pseudorandom generator G in Figure 13. We know that $\mathbf{Adv}^{\mathrm{prg}}_{G,B}(k) = 2 \cdot \Pr\left[\, \text{PRG-DIST}^B_G(k) \Rightarrow 1 \,\right] - 1$. Let PRG-REAL^B_G be the game identical to game PRG-DIST^B_G conditioned on $b = 1$, and PRG-RAND^B_G be the game identical to game PRG-DIST^B_G conditioned on $b = 0$. Then,

$$\mathbf{Adv}^{\mathrm{prg}}_{G,B}(k) = \Pr\left[\, \text{PRG-REAL}^B_G \Rightarrow 1 \,\right] - \Pr\left[\, \text{PRG-RAND}^B_G \Rightarrow 1 \,\right] \ .$$

Algorithm $C(K_G, x)$	**Procedure** $\text{ROSIM}_1(v)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $\mathsf{out} \leftarrow 0$	If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
$pk \leftarrow (K_G, f)$; $b \leftarrow\!\!\$\ \{0,1\}$	Return $H[v]$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	**Procedure** $\text{ROSIM}_2(v)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	If $v = s$ then
$s \leftarrow x \oplus (m_b \| 0^\zeta)$; $t \leftarrow\!\!\$\ \{0,1\}^\rho$	$\mathsf{out} \leftarrow 1$; Halt run of A_2
$c \leftarrow f(s \| t)$	If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
Run $A_2^{\text{ROSIM}_2(\cdot)}(c, state)$	Return $H[v]$
Return out	

Figure 14: **Adversary C in the proof of Theorem 3.3.**

Algorithm $I(f, c)$	**Procedure** $\text{ROSIM}_1(v)$
$b \leftarrow\!\!\$\ \{0,1\}$; $\mathsf{out} \leftarrow \perp$	If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
$K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$; $pk \leftarrow (K_G, f)$	Return $H[v]$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	**Procedure** $\text{ROSIM}_2(v)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	$t \leftarrow\!\!\$\ \mathsf{Ext}(f, c, v)$
Run $A_2^{\text{ROSIM}_2(\cdot)}(c, state)$	If $t \neq \perp$ then
Return out	$\mathsf{out} \leftarrow t$; Halt run of A_2
	If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
	Return $H[v]$

Figure 15: **Inverter I in the proof of Theorem 3.3.**

Note that $\Pr\left[\,\text{PRG-REAL}_G^B \Rightarrow 1\,\right] = \Pr\left[\,G_2(k) \text{ sets } \mathsf{bad}_1\,\right]$. Moreover, in the PRG-RAND_G^B, the probability that any given of adversary A to its RO equals s is $1/2^{\mu+\zeta}$. Taking a union bound over all queries we have $\Pr[\text{PRG-RAND}_G^B \Rightarrow 1] \leq q/2^{\mu+\zeta}$. Thus

$$\Pr\left[\,G_2(k) \text{ sets } \mathsf{bad}_1\,\right] \leq \mathbf{Adv}^{\text{prg}}_{G,B}(k) + \frac{q}{2^{\mu+\zeta}} \ .$$

In game G_3, we reorder the code of game G_2 producing t. The change is conservative, meaning that $\Pr[G_2(k) \Rightarrow 1] = \Pr[G_3(k) \Rightarrow 1]$. Game G_4 is identical to game G_3, except in procedure RO_2. Namely, if adversary A_2 make a query for s, then the oracle lies, calling $\overline{\text{RO}}$ instead. Game G_3 and game G_4 are identical-until-bad_2, thus based on Fundamental Lemma of Game-Playing [14]

$$\Pr\left[\,G_3(k) \Rightarrow 1\,\right] - \Pr\left[\,G_4(k) \Rightarrow 1\,\right] \leq \Pr\left[\,G_4(k) \text{ sets } \mathsf{bad}_2\,\right] \ .$$

Consider the adversary C attacking the pseudorandom generator G in Figure 14. Let PRG-REAL_G^C be the game identical to game PRG-DIST_G^C condition on $b = 1$, and PRG-RAND_G^C be the game identical to game PRG-DIST_G^C condition on $b = 0$. Then,

$$\mathbf{Adv}^{\text{prg}}_{G,C}(k) = \Pr\left[\,\text{PRG-REAL}_G^C \Rightarrow 1\,\right] - \Pr\left[\,\text{PRG-RAND}_G^C \Rightarrow 1\,\right] \ .$$

Note that $\Pr\left[\,\text{PRG-REAL}_G^C \Rightarrow 1\,\right] = \Pr\left[\,G_4(k) \text{ sets } \mathsf{bad}_2\,\right]$. Let Ext be the second-input extractor for $\mathcal{F}$. To bound the probability that PRG-RAND_G^C outputs 1, we construct an inverter I attacking $\mathcal{F}$ in Figure 15. Note that if adversary A_2 queries s then inverter I can invert challenge c using extractor Ext. Hence, we have $\Pr\left[\,\text{PRG-RAND}_G^C \Rightarrow 1\,\right] \leq \mathbf{Adv}^{\text{owf}}_{\mathcal{F},I}(k)$. Thus,

$$\Pr\left[\,G_4(k) \text{ sets } \mathsf{bad}_2\,\right] = \mathbf{Adv}^{\text{prg}}_{G,C}(k) + \mathbf{Adv}^{\text{owf}}_{\mathcal{F},I}(k) \ .$$

Next, game G_5 is identical to game G_4, except it uses a uniformly random x in the encryption phase instead of a pseudorandom value $G(K, r)$. Consider adversary D as shown in Figure 16. We have

$$\Pr[G_4(k) \Rightarrow 1] - \Pr[G_5(k) \Rightarrow 1] \leq \mathbf{Adv}^{\text{prg}}_{G,D}(k) \ .$$

Algorithm $D(K_G, x)$	**Procedure** $\text{ROSIM}_1(v)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
$pk \leftarrow (K_G, f)$; $b \leftarrow\!\!\$\ \{0,1\}$	Return $H[v]$
$(\mathcal{M}_0, \mathcal{M}_1, \text{state}) \leftarrow\!\!\$\ A_1^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	**Procedure** $\text{ROSIM}_2(v)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	If $v = s$ then $H[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
$s \leftarrow x \oplus (m_b \| 0^\zeta)$; $t \leftarrow\!\!\$\ \{0,1\}^\rho$	If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
$c \leftarrow f(s\|t)$	Return $H[v]$
$b' \leftarrow\!\!\$\ A_2^{\text{ROSIM}_2(\cdot)}(c, state)$	
Return $(b = b')$	

Figure 16: **Adversary D in the proof of Theorem 3.3.**

Algorithm $\mathsf{Ext}^H(\mathbf{x}, \mathbf{h}, \mathbf{c}, c_i, pk)$
$(f, K) \leftarrow pk$
For $j = 1$ to $|\mathbf{x}|$ do
  $s_i \| t_i \leftarrow\!\!\$\ \mathsf{Ext}_1(f, c_i, \mathbf{x}[j])$
  If $t_i \neq \perp$ then
    $r_i \leftarrow \mathbf{h}[j] \oplus t_i$; $m_i \leftarrow \mathbf{x}[j] \oplus G(K, r_i)$
    If $m_i|_\zeta = 0^\zeta$ then return $m_i|^\mu$
For $j = 1$ to $|\mathbf{c}|$ do
  $(s_i \| t_i, s_i \| t_i') \leftarrow\!\!\$\ \mathsf{Ext}_2(f, c_i, \mathbf{c}[j])$
  If $s_i \neq \perp$ then
    $r_i \leftarrow H(s_i) \oplus t_i$; $m_i \leftarrow s_i \oplus G(K, r_i)$
    If $m_i|_\zeta = 0^\zeta$ then return $m_i|^\mu$
Return $\perp$

Figure 17: **PA-RO extractor Ext in the proof of Theorem 3.4.**

In game G_6, we reorder the code of game G_5 producing s. The change is conservative, meaning that $\Pr[G_5(k) \Rightarrow 1] = \Pr[G_6(k) \Rightarrow 1]$. Note that $\Pr[G_6(k) \Rightarrow 1] = 1/2$, since the ciphertexts are independent of the bit b. Assuming that the advantage of adversary D is greater than the advantage of B and C, we have

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A}(k) \leq 2 \cdot \mathbf{Adv}^{\text{owf}}_{\mathcal{F},I}(k) + 6 \cdot \mathbf{Adv}^{\text{prg}}_{G,D}(k) + \frac{2q}{2^{\mu+\zeta}} \ .$$

This completes the proof. ∎

PA-RO RESULT. We show RSA-OAEP is PA-RO when modeling $\mathcal{H}$ as a RO if $\mathcal{G}$ is a pseudorandom generator and $\mathcal{F}$ is both second-input extractable and common-input extractable.

Theorem 3.4 Let n, μ, ζ, ρ be integer parameters. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ be a pseudorandom generator and $\mathcal{H} : \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be a RO. Let $\mathcal{F}$ be a family of trapdoor permutations with domain $\{0,1\}^n$, where $n = \mu + \zeta + \rho$. Suppose $\mathcal{F}$ is $(\mu+\zeta)$-second input and $(\mu+\zeta)$-common input extractable. Then $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ is PA-RO secure. In particular, for any adversary A, there exists an adversary D and an extractor Ext such that

$$\mathbf{Adv}^{\text{pa-ro}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A,\mathsf{Ext}}(k) \leq 2 \cdot \mathbf{Adv}^{\text{prg}}_{\mathcal{G},D}(k) + \frac{2q}{2^\zeta} \ .$$

where q is the total number of the extraction queries made by A. Furthermore, the running time of D is about that of A and the running time of Ext is about that of SIE and CIE extractors.

Proof: Let Ext_1 be the second-input extractor and Ext_2 be the common-input extractor for $\mathcal{F}$. For any adversary A, we define the extractor Ext as shown in Figure 17. Now, we bound the advantage of adversary A in distinguishing between the decryption algorithm and the extractor Ext.

Assume adversary A makes q extract queries. Let c_i be the i-th such query. Let's denote by s_i and t_i the last $(\mu+\zeta)$-bits and first ρ-bits of $f^{-1}(c_i)$, respectively. We define S to be the event that game $\text{PA-RO}^{A,\mathsf{Ext}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}]}(k)$

outputs 1. We also define E to be the event that adversary A or the encryption oracle query for the value s_i to random oracle H, for all $i \in [q]$. Then

$$\mathbf{Adv}^{\text{pa-ro}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A,\mathsf{Ext}}(k) = 2 \cdot (\Pr\left[\, S \wedge E \,\right] + \Pr\left[\, S \wedge \overline{E} \,\right]) - 1 \ .$$

We know, if event E happens then extractor Ext can use the second-input extractor or the common-input extractor to recover plaintext m. Therefore, we have $\Pr\left[\, S \mid E \,\right] = 1/2$. Thus,

$$\mathbf{Adv}^{\text{pa-ro}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A,\mathsf{Ext}}(k) \leq \Pr\left[\, E \,\right] + 2 \cdot \Pr\left[\, S \wedge \overline{E} \,\right] - 1 \ .$$

Next, we bound the probability that ciphertext c_i with no prior random oracle query s_i, is valid. We know when event $\overline{E}$ happens, there exists at least a ciphertext c_i with no prior query s_i to random oracle $\mathcal{H}$. Let C be the set of such ciphertexts. Note that extractor Ext always outputs $\perp$ on such ciphertexts. Let T be the event where there exists at least a valid ciphertext $c_i \in C$. Then

$$\begin{aligned} \Pr\left[\, S \wedge \overline{E} \,\right] &= \Pr\left[\, S \wedge \overline{E} \wedge T \,\right] + \Pr\left[\, S \wedge \overline{E} \wedge \overline{T} \,\right] \\ &\leq \Pr\left[\, \overline{E} \wedge T \,\right] + \Pr\left[\, \overline{E} \wedge \overline{T} \,\right] \cdot \Pr\left[\, S \mid \overline{E} \wedge \overline{T} \,\right] \ . \end{aligned}$$

Note that $\Pr\left[\, S \mid \overline{E} \wedge \overline{T} \,\right] = 1/2$, since the outputs of Ext and decryption algorithm Dec are always equal. Moreover, we have $\Pr\left[\, \overline{E} \wedge \overline{T} \,\right] \leq \Pr\left[\, \overline{E} \,\right]$. Therefore,

$$\mathbf{Adv}^{\text{pa-ro}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A,\mathsf{Ext}}(k) \leq 2 \cdot \Pr\left[\, \overline{E} \wedge T \,\right] \ .$$

We know the challenge ciphertext c_i is valid if and only if there exists a plaintext m_i such that $G(K, r_i) \oplus s_i = m_i \| 0^\zeta$, where $r_i = H(s_i) \oplus t_i$. Moreover, when there is no prior random oracle query s_i, r_i looks uniformly random to adversary A. Therefore, if event $T \mid \overline{E}$ happens then there exists a ciphertext c_i such that the first ζ-bits of $\mathcal{G}(K, r_i)$ and s_i are equal, where r_i is chosen uniformly at random. Consider adversary D attacking the pseudorandom generator G in Figure 18. Note that when adversary D is in the real game then it simulates the PA-RO game for the adversary A. On the other hand, when adversary D is in the ideal game then the first ζ-bits of s_i and x are equal with probability $1/2^\zeta$. Taking a union bound, we get

$$\mathbf{Adv}^{\text{prg}}_{\mathcal{G},D}(k) \geq \Pr\left[\, T \wedge \overline{E} \,\right] - \frac{q}{2^\zeta} \ .$$

Hence, we have $\Pr\left[\, T \wedge \overline{E} \,\right] \leq \mathbf{Adv}^{\text{prg}}_{\mathcal{G},D}(k) + q \cdot 2^{-\zeta}$. Summing up,

$$\mathbf{Adv}^{\text{pa-ro}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A,\mathsf{Ext}}(k) \leq 2 \cdot \mathbf{Adv}^{\text{prg}}_{\mathcal{G},D}(k) + \frac{2q}{2^\zeta} \ .$$

This completes the proof. ∎

3.4 Partial Instantiation of H

Now, we instantiate the hash function $\mathcal{H}$ when modeling $\mathcal{G}$ as a RO. In particular, we show $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ is IND-CPA + PA-RO when $\mathcal{H}$ is a special type of hardcore function and $\mathcal{F}$ is one-way, second-input and common-input extractable. Note that Boneh [24] previously showed a simplified RSA-OAEP with one Feistel round $\mathcal{G}$ is IND-CCA2 secure and Barthe *et al.* [3] showed such a scheme does not even need redundancy, but these proofs can not applied to the case of $\mathcal{H}$ as a cryptographic hash function.

IND-CPA RESULT. Under IND-CPA, we show a tight reduction when $\mathcal{H}$ is a $(\mu + \zeta)$-partial hardcore function for $\mathcal{F}$. In particular, it is plausible for $\mathcal{H}$ as a computational randomness extractor [38] and that $\mathcal{F}$ is RSA in the common setting $\rho = k$ (*e.g.*, $\rho = 128$ for modulus length $n = 2048$), since Coppersmith's technique fails.

Theorem 3.5 Let n, μ, ζ, ρ be integer parameters. Let $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be a hash function family and $\mathcal{G} : \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ be a RO. Let $\mathcal{F}$ be a family of trapdoor permutations with domain $\{0,1\}^n$, where

Adversary $D(K, x)$	**Procedure** $\textsc{Enc}(pk, \mathcal{M})$
$\mathsf{out} \leftarrow 0$; $\mathsf{Dom} \leftarrow \perp$	$m \leftarrow\!\!\$\ \mathcal{M}(1^k, pk)$
$i \leftarrow 1$; $j \leftarrow 1$; $b \leftarrow\!\!\$\ \{0,1\}$	$c \leftarrow\!\!\$\ \mathsf{Enc}^{\mathrm{ROSIM}(\cdot,2)}(pk, m)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	$\mathbf{c}[i] \leftarrow c$; $i \leftarrow i + 1$
$pk \leftarrow (K, f)$; $sk \leftarrow (K, f^{-1})$	Return c
$b' \leftarrow\!\!\$\ A^{\mathrm{ROSIM}(\cdot,1),\textsc{Enc}(pk,\cdot),\mathcal{D}(sk,\cdot)}(pk)$	**Procedure** $\mathcal{D}(sk, c)$
Return out	$(s, t) \leftarrow f^{-1}(c)$
Procedure $\mathrm{ROSIM}(s, i)$	If $s \notin \mathsf{Dom}$ then
If $i \neq 3$ then $\mathsf{Dom} \leftarrow \mathsf{Dom} \cup \{s\}$	If $x\|_\zeta = s\|_\zeta$ then $\mathsf{out} \leftarrow 1$
If $H[x] = \perp$ then $H[x] \leftarrow\!\!\$\ \{0,1\}^\ell$	If $c \in \mathbf{c}$ then return $\perp$
If $i = 1$ then	$m_0 \leftarrow \mathsf{Dec}(sk, c)$
$\mathbf{x}[j] \leftarrow x$; $\mathbf{h}[j] \leftarrow H[x]$; $j \leftarrow j+1$	$m_1 \leftarrow\!\!\$\ \mathsf{Ext}^{\mathrm{ROSIM}(\cdot,3)}(\mathbf{x}, \mathbf{h}, \mathbf{c}, c, pk)$
Return $H[x]$	Return m_b

Figure 18: **Adversary D in the proof of Theorem 3.4.**

Games $G_1(k)$, $G_2(k)$	**Games** $G_3(k)$, $G_4(k)$
$b \leftarrow\!\!\$\ \{0,1\}$; $K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$	$b \leftarrow\!\!\$\ \{0,1\}$; $K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_H, f)$	$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_H, f)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{RO}_1(\cdot)}(1^k, pk)$	$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{RO}_1(\cdot)}(1^k, pk)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{RO}_1(\cdot)}(1^k, pk)$; $r \leftarrow\!\!\$\ \{0,1\}^\rho$	$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{RO}_1(\cdot)}(1^k, pk)$
If $G[r] \neq \perp$ then	$r \leftarrow\!\!\$\ \{0,1\}^\rho$; $s \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$
$\mathsf{bad}_1 \leftarrow \mathsf{true}$; $G[r] \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$	$G[r] \leftarrow s \oplus (m_b \| 0^\zeta)$; $z \leftarrow H(K_H, s)$
Else $G[r] \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$	$t \leftarrow z \oplus r$; $c \leftarrow f(s\|t)$
$s \leftarrow G[r] \oplus (m_b \| 0^\zeta)$; $z \leftarrow H(K_H, s)$	$d \leftarrow\!\!\$\ A_2^{\mathrm{RO}_2(\cdot)}(c, state)$
$t \leftarrow z \oplus r$; $c \leftarrow f(s\|t)$	Return $(b = d)$
$d \leftarrow\!\!\$\ A_2^{\mathrm{RO}_2(\cdot)}(c, state)$	**Procedure** $\mathrm{RO}_1(v)$
Return $(b = d)$	Return $\mathrm{RO}(v)$
Procedure $\mathrm{RO}_1(v)$	**Procedure** $\mathrm{RO}_2(v)$
Return $\mathrm{RO}(v)$	If $v = r$ then
Procedure $\mathrm{RO}_2(v)$	$\mathsf{bad}_2 \leftarrow \mathsf{true}$; return $\overline{\mathrm{RO}}(v)$
Return $\mathrm{RO}(v)$	Return $\mathrm{RO}(v)$

Figure 19: **Games G_1–G_4 in the proof of Theorem 3.5.**

$n = \mu + \zeta + \rho$. Suppose $\mathcal{H}$ is a $(\mu + \zeta)$-partial hardcore function for $\mathcal{F}$. Then $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ is IND-CPA. In particular, for any adversary $A = (A_1, A_2)$, there exists an adversary B such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A}(k) \leq 2 \cdot \mathbf{Adv}^{\text{phcf}}_{\mathcal{F},\mathcal{H},B}(k) + \frac{2q}{2^\rho} \ ,$$

where q is the total number of RO queries made by A. The running time of B is about that of A.

Proof: Consider games G_1–G_4 in Figure 19. Each game maintains two independent random oracles RO and $\overline{\mathrm{RO}}$. Procedure RO maintains a local array G as follows:

Procedure $\mathrm{RO}(v)$
If $G[v] = \perp$ then $G[v] \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$
Return $G[v]$

For simplicity, we omit the code of $\mathrm{RO}, \overline{\mathrm{RO}}$ in the games. In each game, we use RO_1 to denote the oracle interface of adversary A_1 and message samplers $\mathcal{M}_0, \mathcal{M}_1$, and we use RO_2 to denote the oracle interface of adversary A_2.

Algorithm $B(K_H, f, c, t, z)$	**Procedure** $\text{ROSIM}_1(v)$
$pk \leftarrow (K_H, f)$; $b \leftarrow\!\!\$\ \{0,1\}$	If $G[v] = \perp$ then $G[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	Return $G[v]$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	**Procedure** $\text{ROSIM}_2(v)$
$r \leftarrow t \oplus z$; $\mathsf{out} \leftarrow \perp$	If $v = r$
Run $A_2^{\text{ROSIM}_2(\cdot)}(c, state)$	$\mathsf{out} \leftarrow 1$; Halt A
Return out	If $G[v] = \perp$ then $G[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
	Return $G[v]$

Figure 20: **Adversary B in the proof of Theorem 3.5.**

Game G_1 corresponds to game $\text{IND-CPA}^A_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}]}$. Then

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A}(k) \le 2 \cdot \Pr[G_1(k) \Rightarrow 1] - 1 \ .$$

We now explain the game chain. Game G_2 is identical to game G_1, except in the encryption of message m_b, if either adversary A_1 or message sampler $\mathcal{M}_b$ queried r to their random oracle RO_1, then it chooses a fresh random value for $G[r]$. Games G_1 and G_2 are identical-until-bad_1, and thus from the Fundamental Lemma of Game-Playing [14],

$$\Pr\left[\, G_1(k) \Rightarrow 1 \,\right] - \Pr\left[\, G_2(k) \Rightarrow 1 \,\right] \le \Pr\left[\, G_2(k) \text{ sets } \mathsf{bad}_1 \,\right] \ .$$

Moreover, the probability is $1/2^\rho$ that any given RO query of A_1 or message samplers $\mathcal{M}_0, \mathcal{M}_1$ equals r . Let q_1 be the number of random oracle query that A_1 and $\mathcal{M}_0, \mathcal{M}_1$ make. Taking a union bound over all queries, we have $\Pr\left[\, G_2(k) \text{ sets } \mathsf{bad}_1 \,\right] \le q_1/2^\rho$. In game G_3, we reorder the code of game G_2 producing s. The change is conservative, meaning that $\Pr[G_2(k) \Rightarrow 1] = \Pr[G_3(k) \Rightarrow 1]$. Game G_4 is identical to game G_3, except in procedure RO_2, if adversary A_2 queries r, then the oracle lies, calling $\overline{\text{RO}}$ instead. Game G_3 and game G_4 are identical-until-bad_2, and based on Fundamental Lemma of Game-Playing [14],

$$\Pr\left[\, G_3(k) \Rightarrow 1 \,\right] - \Pr\left[\, G_4(k) \Rightarrow 1 \,\right] \le \Pr\left[\, G_4(k) \text{ sets } \mathsf{bad}_2 \,\right] \ .$$

Now, consider adversary B attacking partial hardcore function $\mathcal{H}$ in Figure 20. We know that

$$\mathbf{Adv}^{\text{phcf}}_{\mathcal{F},\mathcal{H},B}(k) = 2 \cdot \Pr\left[\, \text{PHCF-DIST}^B_{\mathcal{F},\mathcal{H}}(k) \Rightarrow 1 \,\right] - 1 \ .$$

Let PHCF-REAL be the game identical to game PHCF-DIST conditioned on $b = 1$, and PHCF-RAND be the game identical to game PHCF-DIST conditioned on $b = 0$. Then,

$$\mathbf{Adv}^{\text{phcf}}_{\mathcal{F},\mathcal{H},B}(k) = \Pr\left[\, \text{PHCF-REAL}^B_{\mathcal{F},\mathcal{H}} \Rightarrow 1 \,\right] - \Pr\left[\, \text{PHCF-RAND}^B_{\mathcal{F},\mathcal{H}} \Rightarrow 1 \,\right] \ .$$

Note that $\Pr\left[\, \text{PHCF-REAL}^B_{\mathcal{F},\mathcal{H}} \Rightarrow 1 \,\right] = \Pr\left[\, G_4(k) \text{ sets } \mathsf{bad}_1 \,\right]$. Moreover, in game PHCF-RAND, the probability is $1/2^\rho$ that any given RO queried by adversary A_2 equals r . Let q_2 be the number of queries that A_2 makes. Taking a union bound we have $\Pr[\text{PHCF-RAND}^B_{\mathcal{F},\mathcal{H}} \Rightarrow 1] \le q_2/2^\rho$. Thus

$$\Pr\left[\, G_4(k) \text{ sets } \mathsf{bad}_1 \,\right] \le \mathbf{Adv}^{\text{phcf}}_{\mathcal{F},\mathcal{H},B}(k) + \frac{q_2}{2^\rho} \ .$$

Note that, $\Pr[G_4(k) \Rightarrow 1] = 1/2$, since the distribution of the ciphertexts are completely independent of b. Summing up,

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A}(k) \le 2 \cdot \mathbf{Adv}^{\text{phcf}}_{\mathcal{F},\mathcal{H},B}(k) + \frac{2q}{2^\rho} \ .$$

This completes the proof. ∎

PA-RO RESULT. We show another partial instantiation result modeling $\mathcal{G}$ as a RO. Namely, we show RSA-OAEP is PA-RO if $\mathcal{F}$ is second-input extractable, and common-input extractable. Note that this does not require any

Algorithm $\mathsf{Ext}^H(\mathbf{r}, \mathbf{g}, \mathbf{c}, c_i, pk)$
$(f, K) \leftarrow pk$
For $j = 1$ to $|\mathbf{r}|$ do
  $y_i \leftarrow\!\!\$\ \mathsf{Ext}_1(f, c_i, \mathbf{g}[j]|_\zeta)$
  If $y_i \neq \perp$ then
    $m_i \leftarrow y_i|^{\mu+\zeta} \oplus \mathbf{g}[j]$
    If $m_i|_\zeta = 0^\zeta$ then return $m_i|^\mu$
For $j = 1$ to $|\mathbf{c}|$ do
  $(y_i, y_i') \leftarrow\!\!\$\ \mathsf{Ext}_2(f, c_i, \mathbf{c}[j])$
  $s_i \leftarrow y_i|^{\mu+\zeta}$; $t_i \leftarrow y_i|_\rho$
  If $y_i \neq \perp$ then
    $r_i \leftarrow H(K, s_i) \oplus t_i$; $m_i \leftarrow s_i \oplus G(r_i)$
    If $m_i|_\zeta = 0^\zeta$ then return $m_i|^\mu$
Return $\perp$

Figure 21: **PA-RO extractor Ext in the proof of Theorem 3.6.**

assumption on $\mathcal{H}$.

Theorem 3.6 Let n, μ, ζ, ρ be integer parameters. Let $\mathcal{H} : \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be a hash function family and $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ be a RO. Let $\mathcal{F}$ be a family of trapdoor permutations with domain $\{0,1\}^n$, where $n = \mu+\zeta+\rho$. Suppose $\mathcal{F}$ is $(\rho, \rho+\zeta)$-second input and $(\rho, \rho+\zeta)$-common input extractable. Then $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ is PA-RO secure. In particular, for any adversary A, there exists an extractor Ext such that,

$$\mathbf{Adv}^{\text{pa-ro}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A,\mathsf{Ext}}(k) \leq \frac{2q}{2^\zeta} \ .$$

where q is the total number of the extract queries made by A. The running time of Ext is about that of SIE and CIE extractors.

Proof: Let Ext_1 be the second-input extractor and Ext_2 be the common-input extractor for $\mathcal{F}$. For any adversary A, we define the extractor Ext as shown in Figure 21. Now, we bound the advantage of adversary A in distinguishing between the decryption algorithm and the extractor Ext.

Assume A makes q extract queries. Let c_i be the i-th such query. Let r_i be the randomness used in creating ciphertext c_i. We define S to be the event that game $\text{PA-RO}^{A,\mathsf{Ext}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}]}(k)$ outputs 1. We also define E as the event where A or the encryption oracle query r_i to $\mathcal{G}$, for all $i \in [q]$. Then,

$$\mathbf{Adv}^{\text{pa-ro}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A,\mathsf{Ext}}(k) = 2 \cdot (\Pr[\, S \wedge E \,] + \Pr[\, S \wedge \overline{E} \,]) - 1 \ .$$

We know when the event E happens Ext can use the second-input extractor or the common-input extractor to recover plaintext m. Therefore, we have $\Pr[\, S \mid E \,] = 1/2$. Thus,

$$\mathbf{Adv}^{\text{pa-ro}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A,\mathsf{Ext}}(k) \leq \Pr[\, E \,] + 2 \cdot \Pr[\, S \wedge \overline{E} \,] - 1 \ .$$

Next, we bound the probability that ciphertexts c_i with no prior random oracle query query r_i is valid. We know when event $\overline{E}$ happens, there exists at least one ciphertext c_i with no prior query r_i to random oracle $\mathcal{G}$. Let C be the set of such ciphertexts. Note that extractor Ext always output $\perp$ on such ciphertexts. Let T be the event where there exists at least a valid ciphertext $c_i \in C$. Then,

$$\begin{aligned} \Pr[\, S \wedge \overline{E} \,] &= \Pr[\, S \wedge \overline{E} \wedge T \,] + \Pr[\, S \wedge \overline{E} \wedge \overline{T} \,] \\ &\leq \Pr[\, \overline{E} \wedge T \,] + \Pr[\, \overline{E} \wedge \overline{T} \,] \cdot \Pr[\, S \mid \overline{E} \wedge \overline{T} \,] \ . \end{aligned}$$

Note that $\Pr[\, S \mid \overline{E} \wedge \overline{T} \,] = 1/2$, since the outputs of extractor Ext and decryption algorithm Dec are always

Game $\mathrm{EXT1}_{\mathcal{H}}^{A,\mathcal{E},z}(K_H, r)$	**Procedure** $\mathcal{O}(x_2, y)$
$i \leftarrow 1$; $state \leftarrow \varepsilon$	$(state, x_1) \leftarrow \mathcal{E}(state, K_H, z, x_2, y; r)$
$\mathbf{x} \leftarrow \varepsilon$; $\mathbf{y} \leftarrow \varepsilon$	$\mathbf{x}[i] \leftarrow x_1 \| x_2$; $\mathbf{y}[i] \leftarrow y$
Run $A^{\mathcal{O}(\cdot,\cdot)}(K_H, z; r)$	$i \leftarrow i + 1$
Return $(\mathbf{x}, \mathbf{y})$	Return x_1

Figure 22: **Game to define EXT1 security.**

equal. Moreover, we have $\Pr\left[\,\overline{E} \wedge \overline{T}\,\right] \leq \Pr\left[\,\overline{E}\,\right]$. Therefore,

$$\mathbf{Adv}^{\text{pa-ro}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A,\mathsf{Ext}}(k) \leq 2 \cdot \Pr\left[\,\overline{E} \wedge T\,\right] \ .$$

We know the challenge ciphertext c_i is valid if and only if there exists a plaintext m_i such that $G(r_i) \oplus s_i = m_i \| 0^{\zeta}$. In other words, the challenge ciphertext c_i is a valid ciphertext when $G(r_i)|_{\zeta} \oplus s_i|_{\zeta} = 0^{\zeta}$. Since r_i was not queried, the ciphertext c_i is valid with probability $2^{-\zeta}$. Taking a union bound over all extract queries, we get

$$\mathbf{Adv}^{\text{pa-ro}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A,\mathsf{Ext}}(k) \leq \frac{2q}{2^{\zeta}} \ .$$

This completes the proof. ∎

4 A Hierarchy of Extractability Notions

Intuitively, extractability of a function formalizes the idea that an adversary that produces an image point must "know" a corresponding preimage, as there being a non-blackbox extractor that recovers the preimage. Previous work on extractability starting with [31, 32] considers a "one-shot" adversary. Inspired by PA for encryption schemes [5, 10], we define a hierarchy of EXT for function families, namely EXT0, EXT1, EXT2, and EXT-RO, which will in particular be useful for our full instantiation results. (Even our notion of EXT0 generalizes prior work, as explained below.) However, there are some important differences from PA. First, for EXT the extractor should return the entire preimage whereas in PA the extractor need not return the randomness. Second, PA asks the adversary to distinguish between the answers of the decryption and extraction oracles, while EXT asks the adversary to make the extractor fail to return a preimage.

EXT0 FUNCTIONS. We first give a "one-time" definition of extractability. Let η, ζ, μ be integer parameters. A function family $\mathcal{H} : \mathcal{K}_H \times \mathsf{HDom} \rightarrow \mathsf{HRng}$ is (η, μ)-EXT0$_{\zeta}$ if for any PPT adversary A with coin space Coins, there exists a PPT extractor $\mathcal{E}$ such that, for any key independent auxiliary input $z \in \{0,1\}^{\eta}$:

$$\mathbf{Adv}^{(\eta,\mu)\text{-ext0}_{\zeta}}_{\mathcal{H},A,\mathcal{E},z}(k) = \Pr_{\substack{K_H \xleftarrow{\$} \mathcal{K}_H(1^k) \\ r \xleftarrow{\$} \mathsf{Coins}(k)}} \left[\begin{array}{c} (x_2, y) \leftarrow A(K_H, z; r) \\ \exists x : H(K_H, x)|_{\zeta} = y \ \wedge \ x_2 = x|_{\mu} \end{array} \wedge \begin{array}{c} x_1 \leftarrow \mathcal{E}(K_H, z, x_2, y; r) \\ H(K_H, x_1 \| x_2)|_{\zeta} \neq y \end{array} \right].$$

is negligible in k. We define advantage of A to be $\mathbf{Adv}^{(\eta,\mu)\text{-ext0}_{\zeta}}_{\mathcal{H},A,\mathcal{E}}(k) = \max_{z \in \{0,1\}^{\eta}} \mathbf{Adv}^{(\eta,\mu)\text{-ext0}_{\zeta}}_{\mathcal{H},A,\mathcal{E},z}(k)$.

In other words, the extractor should work when the adversary outputs ζ least significant bits of an image point and μ bits of a preimage, given η bits of auxiliary information. Previous work considered $\zeta = \log|\mathsf{HRng}|$ and $\mu = 0$. Interestingly, considering $\mu > 0$ gives a *non-blackbox* second-input extractability (SIE) notion compared to Section 3.1, which has a black-box notion of SIE. Our non-blackbox notion applies to general function families rather than trapdoor permutations. We also retain the generality afforded by η, ζ, μ below.

Similarly, we define the analogous notion (η, μ)-EXT0$^{\zeta}$ where the adversary outputs the ζ *most* significant bits of the image point. We often write η-EXT0$_{\zeta}$ and η-EXT0$^{\zeta}$ instead of $(\eta, 0)$-EXT0$_{\zeta}$ and $(\eta, 0)$-EXT0$^{\zeta}$, respectively. We also often write (η, μ)-EXT0 instead of (η, μ)-EXT0$_{\zeta}$ when $\zeta = \log|\mathsf{HRng}|$.

EXT1 FUNCTIONS. Next, we give a definition of "many-times" extractability. We note that a central open problem in the theory of extractable functions to construct a "many-times" extractable function from a "one-time" extractable function, see *e.g.* [49]; the obvious approach suffers an extractor "blow-up" issue. For practical purposes, we simply formalize and assume this property for an appropriate construction from cryptographic hashing.

Game $\mathrm{EXT2}^{A,\mathcal{E},z}_{\mathcal{H},\mathcal{F}}(K_H, r)$	**Procedure** $\mathcal{O}(x_2, y)$
$i \leftarrow 1$; $j \leftarrow 1$	If $y \in \mathbf{h}_1$ then return $\perp$
$state \leftarrow \varepsilon$	$(state, x_1) \leftarrow \mathcal{E}(state, K_H, f, z, \mathbf{h}, \mathbf{w}, x_2, y; r)$
$\mathbf{x} \leftarrow \varepsilon$; $\mathbf{y} \leftarrow \varepsilon$	$\mathbf{x}[i] \leftarrow x_1 \| x_2$; $\mathbf{y}[i] \leftarrow y$; $i \leftarrow i + 1$
$\mathbf{h} \leftarrow \varepsilon$; $\mathbf{h}_1 \leftarrow \varepsilon$; $\mathbf{w} \leftarrow \varepsilon$	Return x_1
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	**Procedure** $\mathcal{I}(1^k)$
Run $A^{\mathcal{O}(\cdot,\cdot),\mathcal{I}(\cdot)}(K_H, f, z; r)$	$v \leftarrow\!\!\$\ \mathsf{HDom}(k)$; $h \leftarrow H(K_H, v)$
Return $(\mathbf{x}, \mathbf{y})$	$\mathbf{h}[j] \leftarrow h$; $\mathbf{w}[j] \leftarrow f(v)$; $\mathbf{h}_1[j] \leftarrow h\|_\zeta$
	$j \leftarrow j + 1$
	Return $(h, f(v))$

Figure 23: **Game to define EXT2 security.**

Let η, ζ, μ be integer parameters. Let $\mathcal{H} : \mathcal{K}_H \times \mathsf{HDom} \to \mathsf{HRng}$ be a hash function family. To an adversary A and extractor $\mathcal{E}$, we associate the experiment in Figure 22, for every $k \in \mathbb{N}$. We say $\mathcal{H}$ is (η, μ)-EXT1$_\zeta$ if for any PPT adversary A with coin space Coins, there exists a stateful extractor $\mathcal{E}$ such that, for any key independent auxiliary input $z \in \{0,1\}^\eta$:

$$\mathbf{Adv}^{(\eta,\mu)\text{-ext1}_\zeta}_{\mathcal{H},A,\mathcal{E},z}(k) = \Pr_{\substack{K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k) \\ r \leftarrow\!\!\$\ \mathsf{Coins}(k)}} \left[\begin{array}{c} (\mathbf{x}, \mathbf{y}) \leftarrow \mathrm{EXT1}^{A,\mathcal{E},z}_{\mathcal{H}}(K_H, r) \\ \exists i, \exists x : H(K_H, x)|_\zeta = \mathbf{y}[i] \wedge \mathbf{x}[i]|_\mu = x|_\mu \wedge H(K_H, \mathbf{x}[i])|_\zeta \neq \mathbf{y}[i] \end{array} \right].$$

is negligible in k. We define advantage of A to be $\mathbf{Adv}^{(\eta,\mu)\text{-ext1}_\zeta}_{\mathcal{H},A,\mathcal{E}}(k) = \max_{z \in \{0,1\}^\eta} \mathbf{Adv}^{(\eta,\mu)\text{-ext1}_\zeta}_{\mathcal{H},A,\mathcal{E},z}(k)$. Similarly, we define (η, μ)-EXT1$^\zeta$ where the adversary output ζ most significant bits of the output to be extracted. We often write η-EXT1$_\zeta$ and η-EXT1$^\zeta$ instead of $(\eta, 0)$-EXT1$_\zeta$ and $(\eta, 0)$-EXT1$^\zeta$, respectively.

EXT2 FUNCTIONS. We now extend the definition to give the adversary access to an oracle $\mathcal{I}$ that on input a unary encoding of the security parameter 1^k, outputs a random image along with an uninvertible hint of the corresponding preimage. In other words, this is a form of extractability with key dependent auxiliary information that parallels PA2 for encryption schemes.

Let η, ζ, μ be integer parameters. Let $\mathcal{H} : \mathcal{K}_H \times \mathsf{HDom} \to \mathsf{HRng}$ be a hash function family and $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a trapdoor permutation family with domain HDom. To adversary A and extractor $\mathcal{E}$, we associate the experiment in Figure 23, for every $k \in \mathbb{N}$. We say $\mathcal{H}$ is (η, μ)-EXT2$_\zeta$ with respect to trapdoor permutation family $\mathcal{F}$ if for any PPT adversary A with coin space Coins, there exists a stateful extractor $\mathcal{E}$ such that, for any key independent auxiliary input $z \in \{0,1\}^\eta$:

$$\mathbf{Adv}^{(\eta,\mu)\text{-ext2}_\zeta}_{\mathcal{H},\mathcal{F},A,\mathcal{E},z}(k) = \Pr_{\substack{K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k) \\ r \leftarrow\!\!\$\ \mathsf{Coins}(k)}} \left[\begin{array}{c} (\mathbf{x}, \mathbf{y}) \leftarrow \mathrm{EXT2}^{A,\mathcal{E},z}_{\mathcal{H},\mathcal{F}}(K_H, r) \\ \exists i, \exists x : H(K_H, x)|_\zeta = \mathbf{y}[i] \wedge \mathbf{x}[i]|_\mu = x|_\mu \wedge H(K_H, \mathbf{x}[i])|_\zeta \neq \mathbf{y}[i] \end{array} \right].$$

is negligible in k. The adversary is not allowed to query $y \in \mathbf{h}_1$ for extract oracle $\mathcal{O}$. We define advantage of A to be $\mathbf{Adv}^{(\eta,\mu)\text{-ext2}_\zeta}_{\mathcal{H},\mathcal{F},A,\mathcal{E}}(k) = \max_{z \in \{0,1\}^\eta} \mathbf{Adv}^{(\eta,\mu)\text{-ext2}_\zeta}_{\mathcal{H},\mathcal{F},A,\mathcal{E},z}(k)$. Similarly, we define (η, μ)-EXT2$^\zeta$ where the adversary output ζ most significant bits of the output to be extracted. We often write η-EXT2$_\zeta$ and η-EXT2$^\zeta$ instead of $(\eta, 0)$-EXT2$_\zeta$ and $(\eta, 0)$-EXT2$^\zeta$, respectively.

EXT-RO FUNCTIONS. Finally, we give a notion of extractability in the RO model, inspired by PA-RO for encryption schemes. In particular, here the adversary has access to an oracle $\mathcal{F}$ to which it queries a sampling algorithm, the oracle returning the image of a point in the domain sampled accordingly. Moreover, instead of the adversary's random coins the extractor gets a transcript of its RO queries and responses, but *not* those made by $\mathcal{F}$.

Let ζ be an integer parameter. Let $\mathrm{RO} \colon \{0,1\}^* \to \{0,1\}^*$ be a random oracle and $\mathcal{F} \colon \mathcal{K}_F \times \mathsf{F.Dom} \to \mathsf{F.Rng}$ be a function family. To adversary A and extractor $\mathcal{E}$, we associate the experiment in Figure 22, for every $k \in \mathbb{N}$. We say $\mathcal{F}$ is ζ-EXT-RO if for any PPT adversary A, there exists a PPT extractor $\mathcal{E}$ such that,

$$\mathbf{Adv}^{\zeta\text{-ext-ro}}_{\mathcal{F},A,\mathcal{E}}(k) = \Pr_{K_F \leftarrow\!\!\$\ \mathcal{K}_F(1^k)} \left[\begin{array}{c} (\mathbf{x}, \mathbf{y}) \leftarrow \text{EXT-RO}^{A,\mathcal{E}}_{\mathcal{F}}(K_F) \\ \exists i, \exists x : F(K_F, x)|_\zeta = \mathbf{y}[i] \wedge F(K_F, \mathbf{x}[i])|_\zeta \neq \mathbf{y}[i] \end{array} \right].$$

Game EXT-RO$_{\mathcal{F}}^{A,\mathcal{E}}(K_F)$	**Procedure** $\mathcal{O}(y)$
$i \leftarrow 1$; $j \leftarrow 1$; $p \leftarrow 1$	If $y \in \mathbf{f}$ then return $\perp$
$\mathbf{f} \leftarrow \varepsilon$; $\mathbf{v} \leftarrow \varepsilon$; $\mathbf{h} \leftarrow \varepsilon$	$x \leftarrow \mathcal{E}^{\mathrm{RO}(\cdot,3)}(K_F, \mathbf{f}, \mathbf{v}, \mathbf{h}, y)$
$\mathbf{x} \leftarrow \varepsilon$; $\mathbf{y} \leftarrow \varepsilon$	$\mathbf{x}[i] \leftarrow x$; $\mathbf{y}[i] \leftarrow y$
Run $A^{\mathcal{O}(\cdot),\mathrm{RO}(\cdot,1),\mathcal{F}(\cdot)}(K_F)$	$i \leftarrow i + 1$
Return $(\mathbf{x}, \mathbf{y})$	Return x
Procedure $\mathcal{F}(\mathcal{M})$	**Procedure** $\mathrm{RO}(v, i)$
$m \leftarrow\!\!\$\ \mathcal{M}(1^k)$	If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$\ \{0,1\}^*$
$f \leftarrow F^{\mathrm{RO}(\cdot,2)}(K_F, m)$	If $i = 1$ then
$\mathbf{f}[p] \leftarrow f\|_\zeta$; $p \leftarrow p + 1$	$\mathbf{v}[j] \leftarrow v$; $\mathbf{h}[j] \leftarrow H[v]$; $j \leftarrow j + 1$
Return $f\|_\zeta$	Return $H[v]$

Figure 24: **Game to define EXT-RO security.**

is negligible in k. The adversary is not allowed to query $y \in \mathbf{f}$ for extract oracle $\mathcal{O}$. Note that above, the parameter ζ controls not only what part of the output the adversary queries to its extract oracle, but also what part of the output the *image oracle* returns. This stems from how we use EXT-RO. Namely, we only apply it to the OAEP padding scheme (in Theorem 5.5), which is invertible. As a consequence, in the RO model EXT-RO does not imply EXT2.

Plausibility. We typically use EXT notions in tandem with other properties such as collision-resistance. In terms of feasibility, there are several constructions proposed for EXT0 with $\zeta = \log|\mathsf{HRng}|$ and $\mu = 0$ and collision-resistance in [61] based on knowledge assumptions. (In the weaker case of EXT0 with only one-wayness, which does not suffice for us, the notion is actually achievable for these parameters under standard assumptions [16].) However, for our generalizations and notions of EXT1, EXT2, we are not aware of any constructions in the standard model. Despite the fact that they are difficult to judge, it may be a reality that as a community we need to move to such assumptions in order to make progress on some difficult problems. A similar strategy was used for very different goals by Pandey *et al.* [63]. It would be interesting for future work to explore relations between our assumptions and theirs.

5 Results for Padding Schemes and OAEP

For increased modularity and understanding, we abstract properties of the OAEP padding scheme we need and give some results about how to achieve them based on assumptions on the round functions. That is, at a very high level we would like to show that if $\mathcal{G}$ is xxx-secure and $\mathcal{H}$ is yyy-secure then the OAEP padding scheme is zzz secure. For example, xxx = yyy = zzz = EXT0. Naturally, the actual results, while retaining this flavor, are much more nuanced.

5.1 Scope and Perspective

Parameters and properties. In all properties of the OAEP padding scheme we consider, the adversary produces *part* of the output. In particular, there are two parameter regimes we consider, one where the adversary produces the least-significant $(\zeta + \rho)$-bits of the output (corresponding to t-clear OAEP), and one where the adversary produces the most-significant $(\mu + \rho)$-bits of the output (corresponding to s-clear OAEP). In terms of properties, we consider near-collision resistance, EXT0, EXT1, and EXT-RO. We do not consider EXT2 of the OAEP padding scheme at all.

Plausibility. If we prove a security notion for the OAEP padding scheme based on corresponding assumptions on the round functions, its plausibility reduces to plausibility of the assumptions on the round functions. A case in which we do not know how to do this is that of EXT1 for the "t-clear" parameter regime. In this case, we lend some plausibility to this assumption by showing EXT-RO, which implies EXT1, holds in the RO model. Similar justification was made for assuming a hash function is UCE in [7, Section 6.1].

Adversary $B(K_G)$	**Adversary** $C(K_H)$
$K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$	$K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$
$v_1, v_2 \leftarrow A(K_H, K_G)$	$v_1, v_2 \leftarrow A(K_H, K_G)$
$r_1 \leftarrow v_1\vert_\rho\ ;\ r_2 \leftarrow v_2\vert_\rho$	$m_1 \leftarrow v_1\vert^\mu\ ;\ m_2 \leftarrow v_2\vert^\mu$
If $r_1 \neq r_2$	$r_1 \leftarrow v_1\vert_\rho\ ;\ r_2 \leftarrow v_2\vert_\rho$
Return (r_1, r_2)	If $r_1 \neq r_2$ then return $\perp$
Return $\perp$	$s_1 \leftarrow m_1 \Vert 0^\zeta \oplus G(K_G, r_1)$
	$s_2 \leftarrow m_2 \Vert 0^\zeta \oplus G(K_G, r_2)$
	Return (s_1, s_2)

Figure 25: **Adversaries B and C in the proof of Theorem 5.1.**

5.2 Our Results

Near-collision resistance. We first show that the OAEP padding transform is near-collision resistant wrt. its least-significant bits (*i.e.*, for "t-clear" parameters) if $\mathcal{G}$ is near-collision resistant wrt. its least-significant bits and $\mathcal{H}$ is collision-resistant.

Theorem 5.1 Let μ, ζ, ρ be integer parameters. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be function families. Suppose $\mathcal{G}$ is NCR_ζ and $\mathcal{H}$ is collision resistant. Then $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}]$ is $\mathrm{NCR}_{\zeta+\rho}$. In particular, for any adversary A, there exists adversaries B, C such that

$$\mathbf{Adv}^{\text{n-cr}_{\zeta+\rho}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H}],A}(k) \leq \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G},B}(k) + \mathbf{Adv}^{\text{cr}}_{\mathcal{H},C}(k)\ .$$

The running time of B and C are about that of A.

Proof: Consider near-collision resistance adversary B and collision resistance adversary C in Figure 25. Let v_1, v_2 be the outputs of A. Let S be the event where $v_1 \neq v_2$ are not equal and $\mathsf{OAEP}_{K_G,K_H}(v_1)\vert_{\zeta+\rho} = \mathsf{OAEP}_{K_G,K_H}(v_2)\vert_{\zeta+\rho}$. Let E be the event where the ρ least significant bits of v_1 and v_2 are not equal. Thus,

$$\mathbf{Adv}^{\text{n-cr}_{\zeta+\rho}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H}],A}(k) = \Pr\left[\, S \wedge E \,\right] + \Pr\left[\, S \wedge \overline{E} \,\right]\ .$$

Note that if the event S and E happens the adversary B finds a collision. Thus,

$$\Pr\left[\, S \wedge E \,\right] \leq \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G},B}(k)\ .$$

On the other hand, if $\overline{E}$ happens then the ρ least significant bits of v_1 and v_2 are equal. Moreover, we know v_1 and v_2 are not equal, thus when the event $\overline{E}$ happens the μ most significant bits of v_1 and v_2 are not equal. Therefore, if the event S and $\overline{E}$ happens adversary C finds a collision. Then

$$\Pr\left[\, S \wedge \overline{E} \,\right] \leq \mathbf{Adv}^{\text{cr}}_{\mathcal{H},C}(k)\ .$$

Summing up,

$$\mathbf{Adv}^{\text{n-cr}_{\zeta+\rho}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H}],A}(k) \leq \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G},B}(k) + \mathbf{Adv}^{\text{cr}}_{\mathcal{H},C}(k)\ .$$

This completes the proof. ∎

Theorem 5.2 Let μ, ζ, ρ be integer parameters. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be function families. Suppose $\mathcal{G}$ is NCR_ζ. Then $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}]$ is $\mathrm{NCR}^{\mu+\zeta}$. In particular, for any adversary A, there exists adversary B such that

$$\mathbf{Adv}^{\text{n-cr}^{\mu+\zeta}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H}],A}(k) \leq \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G},B}(k)\ .$$

The running time of B is about that of A.

Proof: Let v_1, v_2 be the outputs of A. Let S be the event where $v_1 \neq v_2$ are not equal and $\mathsf{OAEP}_{K_G,K_H}(v_1)\vert^{\mu+\zeta} = \mathsf{OAEP}_{K_G,K_H}(v_2)\vert^{\mu+\zeta}$. Then, we have $\mathbf{Adv}^{\text{n-cr}^{\mu+\zeta}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H}],A}(k) = \Pr\left[\, S \,\right]$.

Adversary $B(K_G)$
$K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$
$v_1, v_2 \leftarrow A(K_H, K_G)$
$r_1 \leftarrow v_1|_\rho\ ;\ r_2 \leftarrow v_2|_\rho$
Return (r_1, r_2)

Figure 26: **Adversary B in the proof of Theorem 5.2.**

Adversary $A_H(K_H, u; w)$	**Adversary** $A_G(K_G, v; w)$
$\mathsf{out} \leftarrow \perp\ ;\ (K_G, aux) \leftarrow u$	$\mathsf{out} \leftarrow \perp\ ;\ (K_H, aux) \leftarrow v$
Run $A^{\mathcal{O}\text{SIM}(\cdot)}((K_G, K_H), aux; w)$	Run $A^{\mathcal{O}\text{SIM}(\cdot)}((K_G, K_H), aux; w)$
Return (s_2, out)	Return $(\perp, \mathsf{out})$
Procedure $\mathcal{O}\text{SIM}(y)$	**Procedure** $\mathcal{O}\text{SIM}(y)$
$(s_2, t) \leftarrow y\ ;\ v \leftarrow (K_H, aux)$	$(s_2, t) \leftarrow y$
$r \leftarrow \mathsf{Ext}_G(K_G, v, \perp, s_2; w)$	$\mathsf{out} \leftarrow s_2$
$\mathsf{out} \leftarrow r \oplus t$	Halt A
Halt A	

Figure 27: **Adversaries A_H, A_G in the proof of Theorem 5.3.**

Note that when $\mathsf{OAEP}_{K_G,K_H}(v_1)|^{\mu+\zeta} = \mathsf{OAEP}_{K_G,K_H}(v_2)|^{\mu+\zeta}$, we have $G(K_G, v_1|_\rho)|_\zeta = G(K_G, v_2|_\rho)|_\zeta$. Moreover, since $\mathsf{OAEP}_{K_G,K_H}(v_1)|^{\mu+\zeta} = \mathsf{OAEP}_{K_G,K_H}(v_2)|^{\mu+\zeta}$, if $v_1|_\rho$ equals to $v_2|_\rho$, v_1 will be equal to v_2. Consider near-collision resistance adversary B in Figure 26. When adversary A finds a near collision, B also finds a near collision. Therefore we have $\Pr[S] \le \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G},B}(k)$. Summing up,

$$\mathbf{Adv}^{\text{n-cr}^{\mu+\zeta}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H}],A}(k) \le \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G},B}(k)\ .$$

This completes the proof. ∎

EXT0 RESULT. In more detail, we show that the OAEP padding transform is EXT0 wrt. its least-significant bits (*i.e.*, for "t-clear" parameters) if $\mathcal{G}$ is EXT0 wrt. its least significant bits and injective, and $\mathcal{H}$ is also suitably EXT0. Namely, for $\mathcal{H}$ the extractor gets the image point and ζ-bits of preimage, so since $\mathcal{H}$ maps $\{0,1\}^{\zeta+\mu}$ to ρ, if $\mu \ll \rho$ this assumption would be reasonable.

Theorem 5.3 Let $\eta, \delta, \lambda, \mu, \zeta, \rho$ be integer parameters. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be function families. Let $\eta = |[\mathcal{K}_H(1^k)]| + \lambda$ and $\delta = |[\mathcal{K}_G(1^k)]| + \lambda$. Suppose $\mathcal{G}$ is η-EXT0$_\zeta$, ζ-injective and $\mathcal{H}$ is (δ,ζ)-EXT0. Then, $\mathsf{OAEP}[\mathcal{G},\mathcal{H}]$ is λ-EXT0$_{\zeta+\rho}$. In particular, for any EXT0 adversary A, there exist EXT0 adversaries A_G, A_H and an extractor Ext such that for all extractors $\mathsf{Ext}_G, \mathsf{Ext}_H$

$$\mathbf{Adv}^{\lambda\text{-ext0}_{\zeta+\rho}}_{\mathsf{OAEP},A,\mathsf{Ext}}(k) \le \mathbf{Adv}^{\eta\text{-ext0}_\zeta}_{\mathcal{G},A_G,\mathsf{Ext}_G}(k) + \mathbf{Adv}^{(\delta,\zeta)\text{-ext0}}_{\mathcal{H},A_H,\mathsf{Ext}_H}(k)\ .$$

The running time of A_G is about that of A. The running time of A_H is about that of A plus the time to run Ext_G. The running time of Ext is the time to run Ext_G and Ext_H.

Proof: Let w be the randomness of adversary A, and let $aux \in \{0,1\}^\lambda$ be the key-independent auxiliary input to the adversary A in the game EXT0. Let K_G and K_H be the keys for function families $\mathcal{G}$ and $\mathcal{H}$ respectively. Let $v = (K_H, aux)$ be the key-independent auxiliary input to A_G in the game EXT0, and let $u = (K_G, aux)$ be the key-independent auxiliary input to A_H in the game EXT0. Note that auxiliary input v and u are independent of keys K_G and K_H, respectively. We define adversaries A_G, A_H with random coins w in Figure 27. Let Ext_G and Ext_H be the corresponding extractor for A_G and A_H, respectively. We define EXT0 extractor Ext as shown in Figure 28.

Note that for the extract query y that A makes, if y is not a valid image point then extractor Ext outputs $\perp$. Thus, adversary A does not win by making an invalid image query. Hence, we assume wlog that A only queries a valid image point y. Let $m_y \| r_y$ be the corresponding input for y and $s_y = G(K_G, r_y) \oplus (m_y \| 0^\zeta)$ be the intermediate value. Let r be the output of Ext_G and s_1 be the output of Ext_H. Wlog, we can assume when extractors Ext_G and Ext_H output a non-empty string, they were successful in finding preimages.

Algorithm $\mathsf{Ext}((K_G, K_H), aux, \perp, y; w)$
$(s_2, t) \leftarrow y$; $v \leftarrow (K_H, aux)$; $u \leftarrow (K_G, aux)$
$r \leftarrow \mathsf{Ext}_G(K_G, v, \perp, s_2; w)$; $z \leftarrow r \oplus t$
$s_1 \leftarrow \mathsf{Ext}_H(K_H, u, s_2, z; w)$; $s \leftarrow s_1 \| s_2$
$m^* \leftarrow s \oplus G(K_G, r)$; $m \leftarrow m^*|^\mu$
If $m^*|_\zeta \neq 0^\zeta$ then return $\perp$
If $y \neq \mathsf{OAEP}_{(K_G, K_H)}(m\|r)|_{\zeta+\rho}$ then return $\perp$
Return $m\|r$

Figure 28: **EXT0 extractor Ext in the proof of Theorem 5.3.**

Adversary $A_G^{\mathcal{O}_G(\cdot)}(K_G, v; w)$
$(K_H, aux) \leftarrow v$
Run $A^{\mathcal{O}_{\mathsf{OAEP}}\textsc{Sim}(\cdot)}(K_G, K_H, aux; w)$

Procedure $\mathcal{O}_{\mathsf{OAEP}}\textsc{Sim}(y)$
$r \leftarrow \mathcal{O}_G(y|_\zeta)$
$x \leftarrow G(K_G, r)$
$m^* \leftarrow y \oplus x$; $m \leftarrow m^*|^\mu$
Return $m\|r$

Figure 29: **EXT1 adversary A_G in the proof of Theorem 5.4.**

Consider EXT0 adversaries A_G, A_H in Figure 27. We know A always makes a valid image query. Thus, A wins if extractor Ext outputs $\perp$. On the other hand, Ext outputs $\perp$ only if either Ext_G or Ext_H fails. Moreover, we know if Ext_G outputs a non-empty string, then it will return $r = r_y$, since $\mathcal{G}$ is ζ-injective. Therefore,

$$\mathbf{Adv}^{\lambda\text{-ext0}_{\zeta+\rho}}_{\mathsf{OAEP},A,\mathsf{Ext}}(k) \leq \mathbf{Adv}^{\eta\text{-ext0}_\zeta}_{\mathcal{G},A_G,\mathsf{Ext}_G}(k) + \mathbf{Adv}^{(\delta,\zeta)\text{-ext0}}_{\mathcal{H},A_H,\mathsf{Ext}_H}(k) \ .$$

This completes the proof. ∎

EXT1 RESULT. We next show that the OAEP padding transform is EXT1 wrt. its most-significant bits (*i.e.*, "s-clear" parameters) when $\mathcal{G}$ is EXT1 wrt. its least-significant bits. Note this does not use any assumption on $\mathcal{H}$.

Theorem 5.4 Let $\eta, \lambda, \mu, \zeta, \rho$ be integer parameters. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be function families. Let $\eta = |[\mathcal{K}_H(1^k)]| + \lambda$. Suppose $\mathcal{G}$ is η-EXT1$_\zeta$. Then, $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}]$ is λ-EXT1$^{\mu+\zeta}$. In particular, for any EXT1 adversary A, there exists an EXT1 adversary A_G and an extractor Ext such that for all extractor Ext_G

$$\mathbf{Adv}^{\lambda\text{-ext1}^{\mu+\zeta}}_{\mathsf{OAEP},A,\mathsf{Ext}}(k) \leq \mathbf{Adv}^{\eta\text{-ext1}_\zeta}_{\mathcal{G},A_G,\mathsf{Ext}_G}(k) \ .$$

The running time of A_G is about that of A and the running time of Ext is about that of Ext_G.

Proof: Let w be the randomness of adversary A, $aux \in \{0,1\}^\lambda$ be the key independent auxiliary input to adversary A in the game EXT1. Let K_G and K_H be the key for the function family $\mathcal{G}$ and $\mathcal{H}$ respectively. Let $v = (K_H, aux)$ be the key independent auxiliary input to A_G in the game EXT1. We define adversary A_G with the randomness w in Figure 29. Let Ext_G be the corresponding extractor for A_G. We also define EXT1 extractor Ext as shown in Figure 30.

Note that for every extract query y that A makes, if y is not valid then extractor Ext outputs $\perp$. Thus, adversary A does not win by making an invalid image query. Hence, we assume wlog that the adversary A only queries for the valid images.

We define S to be the event that extractor Ext outputs empty string on at least one of the extract queries. Note that, since we assume that A queries only for the valid images then $\mathbf{Adv}^{\lambda\text{-ext1}^{\mu+\zeta}}_{\mathsf{OAEP},A,\mathsf{Ext}}(k) = \Pr[S]$. Moreover, we know Ext output empty string only if Ext_G outputs empty string on valid image query. Thus, we have $\Pr[S] \leq \mathbf{Adv}^{\eta\text{-ext1}_\zeta}_{\mathcal{G},A_G,\mathsf{Ext}_G}(k)$.

Algorithm $\mathsf{Ext}(state, (K_G, K_H), aux, \bot, y; w)$
$v \leftarrow (K_H, aux)$
$(st, r) \leftarrow \mathsf{Ext}_G(st, K_G, v, \bot, y|_\zeta; w)$
$state \leftarrow st$; $x \leftarrow G(K_G, r)$
$m^* \leftarrow y \oplus x$; $m \leftarrow m^*|^\mu$
If $m^*|_\zeta \neq 0^\zeta$ then return $(state, \bot)$
If $y \neq \mathsf{OAEP}_{(K_G,K_H)}(m\|r)|^{\mu+\zeta}$ then return $(state, \bot)$
Return $(state, m\|r)$

Figure 30: **EXT1 extractor Ext in the proof of Theorem 5.4.**

Algorithm $\mathsf{Ext}^{\mathcal{G},\mathcal{H}}(\bot, \mathbf{y}, (\mathbf{r}, \mathbf{s}), (\mathbf{x}, \mathbf{z}), y)$
$r \leftarrow \bot$; $s \leftarrow \bot$
For $i = 1$ to $|\mathbf{r}|$ do
  If $y|^\zeta = \mathcal{G}(\mathbf{r}[i])|_\zeta$
    For $j = 1$ to $|\mathbf{s}|$ do
      If $\mathcal{H}(\mathbf{s}[j]) = \mathbf{r}[i] \oplus y|_\rho$
        $s \leftarrow \mathbf{s}[j]$; $r \leftarrow \mathbf{r}[i]$
If $(r = \bot \vee s = \bot)$ then return $\bot$
$m^* \leftarrow s \oplus \mathcal{G}(r)$; $m \leftarrow m^*|^\mu$
Return $m\|r$

Figure 31: **EXT-RO extractor Ext in the proof of Theorem 5.5.**

Summing up,

$$\mathbf{Adv}^{\lambda\text{-ext1}^{\mu+\zeta}}_{\mathsf{OAEP},A,\mathsf{Ext}}(k) \leq \mathbf{Adv}^{\eta\text{-ext1}_\zeta}_{\mathcal{G},A_G,\mathsf{Ext}_G}(k) \ .$$

This completes the proof. ∎

EXT-RO RESULT. We would also like to show that the OAEP padding transform is EXT1 wrt. its least-significant bits (*i.e.*, "t-clear" parameters) but we are unable to do so. (The straightforward approach has an "extractor blow-up" problem.) To lend plausibility to this assumption, we instead turn to the RO model and show that the OAEP padding transform is EXT-RO (which implies EXT1) if $\mathcal{G}$ and $\mathcal{H}$ are modeled as ROs.

Theorem 5.5 Let $\mathcal{G} : \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ and $\mathcal{H} : \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be ROs. Then $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}]$ is $(\zeta + \rho)$-EXT-RO. In particular, for any adversary A, there exists an extractor Ext such that,

$$\mathbf{Adv}^{(\zeta+\rho)\text{-ext-ro}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H}],A,\mathsf{Ext}}(k) \leq q_1 \cdot 2^{\mu-\rho} + q_1 \cdot 2^{\mu-\zeta} + \frac{q_1(q_2+q_3)}{2^\zeta} + \frac{q_1(q_2+q_3)^2}{2^\zeta} \ .$$

where q_1 is the total number of extract queries, q_2 is the total number of image oracle queries and q_3 is the total number of random oracle queries made by A.

Proof: For any adversary A, we define the extractor Ext as shown in Figure 31. Now, we bound the probability that extractor Ext fails on at least one of the extract queries made by adversary A. We define $R_\mathcal{I}$ and $S_\mathcal{I}$ to be the set of queries made by image oracle to $\mathcal{G}$ and $\mathcal{H}$, respectively. We also define R_A and S_A to be the set of queries made by A to $\mathcal{G}$ and $\mathcal{H}$, respectively. Let $R = R_\mathcal{I} \cup R_A$ and $S = S_\mathcal{I} \cup S_A$. We define C to be the event where there exists $r_1, r_2 \in R$ such that $\mathcal{G}(r_1)|_\zeta = \mathcal{G}(r_2)|_\zeta$. Note that we have $\Pr[C] \leq (q_2+q_3)^2/2^\zeta$. Let y_i be i-th extract query made by A, for all $i \in [q_1]$. For all $i \in [q_1]$, we define S_i to be the event where y_i is a valid image and extractor Ext outputs $\bot$ on input y_i. Then,

$$\mathbf{Adv}^{(\zeta+\rho)\text{-ext-ro}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H}],A,\mathsf{Ext}}(k) \leq \Pr[C] + \sum_i \Pr[S_i \wedge \overline{C}] \ .$$

We define R' to be the set of all $r \in R$ where $\mathcal{G}(r)|_\zeta = y_i|^\zeta$. Note that when $\overline{C}$ happens, there are no collision on set R and we get that $|R'| \leq 1$. We define E to be the event where $|R'| = 0$. Note that when E happens, challenge

Game $\$\text{SIM-CPA-KI-REAL}^{A,\mathcal{M}}_{\mathsf{PKE}}(k)$	**Game** $\$\text{SIM-CPA-KI-IDEAL}^{A,S,\mathcal{M}}_{\mathsf{PKE}}(k)$
$param \leftarrow\!\!\$\ A.\text{pg}(1^k)$	$param \leftarrow\!\!\$\ A.\text{pg}(1^k)$
$(pk, sk) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	$(pk, sk) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$
$\mathbf{m} \leftarrow\!\!\$\ \mathcal{M}(1^k, param)$	$\mathbf{m} \leftarrow\!\!\$\ \mathcal{M}(1^k, param)$
$\mathbf{c} \leftarrow \mathsf{Enc}(pk, \mathbf{m})$	$\omega \leftarrow\!\!\$\ S(pk, \lvert\mathbf{m}\rvert, param)$
$\omega \leftarrow\!\!\$\ A.\text{g}(pk, \mathbf{c}, param)$	Return $(\omega = A.\text{f}(\mathbf{m}, param))$
Return $(\omega = A.\text{f}(\mathbf{m}, param))$	

Figure 32: **Games to define $SIM-CPA-KI security.**

y_i is a valid image if there exist $s \in S$ such that $s|_\zeta = y_i|^\zeta$ and $\mathcal{G}(\mathcal{H}(s) \oplus y_i|_\rho)|_\zeta = y_i|^\zeta$ or if there exist $s \notin S$ such that $s|_\zeta = y_i|^\zeta$ and $\mathcal{G}(\mathcal{H}(s) \oplus y_i|_\rho)|_\zeta = y_i|^\zeta$. Therefore, we obtain that $\Pr\left[\, S \wedge \overline{C} \wedge E \,\right] \leq 2^{\mu-\zeta} + (q_2 + q_3)/2^\zeta$.

Let $Z = \{z = r \oplus y_i|_\rho : \forall r \in R'\}$. We know when $\overline{C}$ and $\overline{E}$ happens, size of the set Z is equal to 1. Let z be such a element in Z. Let $S' = \{s \in S : \mathcal{H}(s) = z \wedge s|_\zeta = y_i|^\zeta\}$. We define T to be the event where S' is empty. Note that when $\overline{C}$, $\overline{E}$ and T happens, challenge y_i is a valid image if there exist s such that $\mathcal{H}(s) = z$. Therefore, we obtain that $\Pr\left[\, S \wedge \overline{C} \wedge \overline{E} \wedge T \,\right] \leq 2^{\mu-\rho}$.

Note that when $\overline{T}$ happens, we have two cases. First, we have that $S_A \cap S'$ is non-empty. Note that when $S_A \cap S'$ is non-empty, extractor Ext can extract the correct preimage. Next, we have that $S_A \cap S'$ is empty. Then, we know that $S_{\mathcal{I}} \cap S'$ is non-empty. Note that when $S_{\mathcal{I}} \cap S'$ is non-empty, we get collision on $\mathcal{G}$ if challenge y_i is valid image. Thus, we obtain that $\Pr\left[\, S \wedge \overline{C} \wedge \overline{E} \wedge \overline{T} \,\right] = 0$.

Summing up,

$$\mathbf{Adv}^{(\zeta+\rho)\text{-ext-ro}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H}],A,\mathsf{Ext}}(k) \leq q_1 \cdot 2^{\mu-\rho} + q_1 \cdot 2^{\mu-\zeta} + \frac{q_1(q_2+q_3)}{2^\zeta} + \frac{q_1(q_2+q_3)^2}{2^\zeta} \ .$$

This completes the proof. ∎

6 Full Instantiation Results for t-Clear RSA-OAEP

In this section, we give full instantiation results for t-clear RSA-OAEP.

6.1 Notions of High-Entropy-Message Security

We define notions of security for encryption high-entropy messages. Previously, security for high entropy messages has been considered for *deterministic* encryption (*i.e.*, where the encryption algorithm is deterministic) [4], but it makes sense for (public-key) randomized encryption as well. Indeed in the symmetric-key, information theoretic setting such security was considered by [42]. We show an analogous equivalence of indistinguishability and semantic security in this setting. As for deterministic encryption [4] and prior full instantiations of t-clear RSA-OAEP [22], we consider messages independent of the public key. However, unlike for deterministic encryption, this independence is not inherent for randomized encryption. We also note that we could avoid the high min-entropy requirement by using the assumption that $\mathcal{H}$ and $\mathcal{G}$ are UCE and applying the result of [7], but we prefer to stick with our milder assumptions.

Induced distributions. Let X, X' be distributions (or random variables) on the same domain. For $\alpha \in \mathbb{N}$, we say that X' is an *α-induced distribution* [47] of X if X' is a conditional distribution and $X' = X|E$ for an event E such that $\Pr[\, E \,] \geq 2^{-\alpha}$. We call E the corresponding event to X'. We require that the joint distribution (X, E) is efficiently sampleable where we view event E as a binary random variable.

High-entropic message samplers. A *message sampler* $\mathcal{M}$ is a PPT algorithm that takes as input 1^k and a string $param \in \{0,1\}^*$, and outputs a vector $\mathbf{m}$ of messages. A message sampler $\mathcal{M}$ is (ℓ, v)-*entropic* if for any $k \in \mathbb{N}$, any $param \in \{0,1\}^*$, and $\mathbf{m} \leftarrow\!\!\$\ \mathcal{M}(1^k, param)$, we have $|\mathbf{m}| = v$ and each message $\mathbf{m}[i]$ (with $i \in \{1, \ldots, v\}$) must have min-entropy at least ℓ. We require that $\mathcal{M}$ be associated with function $n(\cdot)$ such that for any $param \in \{0,1\}^*$, for any $k \in \mathbb{N}$, and any $\mathbf{m} \in [\mathcal{M}(1^k, param)]$, we have $|\mathbf{m}[i]| = n(k)$, for every $i \leq |\mathbf{m}|$.

$SIM-CPA-KI security. Let $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE scheme. To a message sampler $\mathcal{M}$ and an adversary A and simulator S, we associate the experiment in Figure 32, for every $k \in \mathbb{N}$. We say that PKE is

Game $\text{\$IND-CPA-KI}^{A,\mathcal{M}_0,\mathcal{M}_1}_{\mathsf{PKE}}(k)$
$b \leftarrow\!\!\$ \{0,1\}$; $param \leftarrow\!\!\$ A.\text{pg}(1^k)$
$(pk, sk) \leftarrow\!\!\$ \mathsf{Kg}(1^k)$
$\mathbf{m} \leftarrow\!\!\$ \mathcal{M}_b(1^k, param)$
$\mathbf{c} \leftarrow \mathsf{Enc}(pk, \mathbf{m})$
$b' \leftarrow\!\!\$ A.\text{g}(pk, \mathbf{c}, param)$
Return $(b = b')$

Figure 33: **Games to define $IND-CPA-KI security.**

$SIM-CPA-KI secure for a class $\mathscr{M}$ of message samplers if for every $\mathcal{M} \in \mathscr{M}$ and any adversary A there exists a simulator S,

$$\begin{aligned}\mathbf{Adv}^{\text{\$sim-cpa-ki}}_{\mathsf{PKE},A,S,\mathcal{M}}(k) &= \Pr\left[\,\text{\$SIM-CPA-KI-REAL}^{A,\mathcal{M}}_{\mathsf{PKE}}(k) \Rightarrow 1\,\right] \\ &\quad - \Pr\left[\,\text{\$SIM-CPA-KI-IDEAL}^{A,S,\mathcal{M}}_{\mathsf{PKE}}(k) \Rightarrow 1\,\right] \ .\end{aligned}$$

is negligible in k.

CCA1 extension. To add a CCA1 flavor to $SIM-CPA-KI, a notion which we call $SIM-CCA1-KI, one would allow adversary A.pg and message sampler $\mathcal{M}$ oracle access to $\mathsf{Dec}(sk, \cdot)$. Let $SIM-CCA1-KI-REAL and $SIM-CCA1-KI-IDEAL be the corresponding experiments, and define

$$\begin{aligned}\mathbf{Adv}^{\text{\$sim-cca1-ki}}_{\mathsf{PKE},A,S,\mathcal{M}}(k) &= \Pr\left[\,\text{\$SIM-CCA1-KI-REAL}^{A,\mathcal{M}}_{\mathsf{PKE}}(k) \Rightarrow 1\,\right] \\ &\quad - \Pr\left[\,\text{\$SIM-CCA1-KI-IDEAL}^{A,S,\mathcal{M}}_{\mathsf{PKE}}(k) \Rightarrow 1\,\right] \ .\end{aligned}$$

We say that PKE is $SIM-CCA1-KI secure for a class $\mathscr{M}$ of message samplers if for every $\mathcal{M} \in \mathscr{M}$ and any adversary A there exists a simulator S, such that $\mathbf{Adv}^{\text{\$sim-cca1-ki}}_{\mathsf{PKE},A,S,\mathcal{M}}(k)$ is negligible in k.

$IND-CPA-KI security. Let $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE scheme. To message samplers $\mathcal{M}_0, \mathcal{M}_1$ and an adversary A, we associate the experiment in Figure 33, for every $k \in \mathbb{N}$. We say that PKE is $IND-CPA-KI secure for a class $\mathscr{M}$ of message samplers if for every $\mathcal{M}_0, \mathcal{M}_1 \in \mathscr{M}$ and any adversary A,

$$\mathbf{Adv}^{\text{\$ind-cpa-ki}}_{\mathsf{PKE},A,\mathcal{M}_0,\mathcal{M}_1}(k) = 2 \cdot \Pr\left[\,\text{\$IND-CPA-KI}^{A,\mathcal{M}_0,\mathcal{M}_1}_{\mathsf{PKE}}(k) \Rightarrow 1\,\right] - 1 \ .$$

is negligible in k.

CCA1 extension. To add a CCA1 flavor to $IND-CPA-KI, a notion which we call $IND-CCA1-KI, one would allow adversary A.pg and message samplers $\mathcal{M}_0, \mathcal{M}_1$ oracle access to $\mathsf{Dec}(sk, \cdot)$. Let $IND-CCA1-KI be the corresponding experiment, and define

$$\mathbf{Adv}^{\text{\$ind-cca1-ki}}_{\mathsf{PKE},A,\mathcal{M}_0,\mathcal{M}_1}(k) = 2 \cdot \Pr\left[\,\text{\$IND-CCA1-KI}^{A,\mathcal{M}_0,\mathcal{M}_1}_{\mathsf{PKE}}(k) \Rightarrow 1\,\right] - 1 \ .$$

We say that PKE is $IND-CCA1-KI secure for a class $\mathscr{M}$ of message samplers if for every $\mathcal{M}_0, \mathcal{M}_1 \in \mathscr{M}$ and any adversary A, $\mathbf{Adv}^{\text{\$ind-cca1-ki}}_{\mathsf{PKE},A,\mathcal{M}_0,\mathcal{M}_1}(k)$ is negligible in k.

Other definitions. We replace CCA1 with CCA0 above when the adversary makes only one decryption query. We do not extend the definition to CCA2 because we do not use it, but this can be done in the natural way.

Definitional equivalence. We show in the Appendix C that $IND-CPA-KI implies $SIM-CPA-KI. The proof is similar to the proof of Theorem 3.1 from [47]. Similarly, we have that $IND-CCA1-KI implies $SIM-CCA1-KI. Therefore we use the former in our results. We omit the reverse implication for simplicity.

Single versus multi-message security. We note that the $IND-CPA-KI security for a single message does not imply $IND-CPA-KI security for multiple messages in general. The proof is similar to that in [4]. Thus, in our results, we explicitly use multiple messages.

6.2 Main Results

We show that t-clear RSA-OAEP is \$IND-CCA0-KI and \$IND-CCA1-KI under respective suitable assumptions. As in Section 3 we actually prove corresponding notions of IND-CPA + PA, yielding stronger results. The results in follow from those below combined.

\$IND-CCA0-KI RESULT. Interestingly, for \$IND-CCA0-KI we use milder assumptions on $\mathcal{G}$ by performing a direct analysis of OAEP rather than abstracting a property of the underlying the padding scheme and applying the results of Section 5. Namely, we avoid the assumption that $\mathcal{G}$ is ζ-injective.

Theorem 6.1 Let $\eta, \delta, \mu, \zeta, \rho$ be integer parameters. Let $\mathcal{F}$ be a family of one-way trapdoor permutations with domain $\{0,1\}^\mu$. Let $\mathscr{M}$ be a class of (ℓ, v)-entropic message samplers, $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be function families. Let $\eta = |[\mathsf{Kg}(1^k)]| + |[\mathcal{K}_H(1^k)]|$ and $\delta = |[\mathsf{Kg}(1^k)]| + |[\mathcal{K}_G(1^k)]|$. Suppose $\mathcal{G}$ is η-EXT0$_\zeta$ and NCR$_\zeta$, $\mathcal{H}$ is (δ,ζ)-EXT0 and CR. Also suppose $\mathcal{G}$ is a pseudorandom generator and $\mathcal{H}$ is a hardcore function for $\mathcal{F}|_\zeta$ on class $\mathscr{M}$. Then $\mathsf{OAEP}_{\mathsf{t\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|_{\zeta+\rho}]$ is \$IND-CCA0-KI secure. In particular, for every $\mathcal{M}_0, \mathcal{M}_1 \in \mathscr{M}$ and any adversary A, there exists adversaries A_G, A_H, B_G, B_H, B, C and a distribution $\mathbf{X}(k) \in \mathscr{M}$ such that for all extractors $\mathsf{Ext}_G, \mathsf{Ext}_H$

$$\begin{aligned}\mathbf{Adv}^{\text{\$ind-cca0-ki}}_{\mathsf{OAEP}_{\mathsf{t\text{-}clear}},A,\mathcal{M}_0,\mathcal{M}_1}(k) \;\leq\; & 2\cdot\mathbf{Adv}^{\eta\text{-ext0}_\zeta}_{\mathcal{G},A_G,\mathsf{Ext}_G}(k) + 2\cdot\mathbf{Adv}^{(\delta,\zeta)\text{-ext0}}_{\mathcal{H},A_H,\mathsf{Ext}_H}(k) \\ & +2\cdot\mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G},B_G}(k) + 2\cdot\mathbf{Adv}^{\text{cr}}_{\mathcal{H},B_H}(k) + 2\cdot\mathbf{Adv}^{\text{hcf}}_{\mathcal{F}|_\zeta,\mathcal{H},\mathbf{X},B}(k) + 2v\cdot\mathbf{Adv}^{\text{prg}}_{\mathcal{G},C}(k)\ .\end{aligned}$$

\$IND-CCA1-KI RESULT. To prove \$IND-CCA1-KI, we use EXT1 and near-collision resistance of the overall OAEP padding scheme (which follows from corresponding assumptions on the round functions as per Section 5), as well as the assumption that $\mathcal{G}$ is a pseudorandom generator and $\mathcal{H}$ is an appropriate hardcore function.

Theorem 6.2 Let η, μ, ζ, ρ be integer parameters. Let $\mathcal{F}$ be a family of one-way trapdoor permutations with domain $\{0,1\}^\mu$. Let $\mathscr{M}$ be a class of (ℓ, v)-entropic message samplers and $\eta = |[\mathsf{Kg}(1^k)]|$. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be function families. Suppose $\mathcal{G}$ is a pseudorandom generator, and let $\mathcal{H}$ be a hardcore function for $\mathcal{F}|_\zeta$ on class $\mathscr{M}$. Also suppose $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}]$ is η-EXT1$_{\zeta+\rho}$ and NCR$_{\zeta+\rho}$. Then $\mathsf{OAEP}_{\mathsf{t\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|_{\zeta+\rho}]$ is \$IND-CCA1-KI secure. In particular, for every $\mathcal{M}_0, \mathcal{M}_1 \in \mathscr{M}$ and any adversary A, there exists adversaries B, C, D, E and a distribution $\mathbf{X}(k) \in \mathscr{M}$ such that for all extractors Ext

$$\begin{aligned}\mathbf{Adv}^{\text{\$ind-cca1-ki}}_{\mathsf{OAEP}_{\mathsf{t\text{-}clear}},A,\mathcal{M}_0,\mathcal{M}_1}(k) \;\leq\; & 2\cdot\mathbf{Adv}^{\eta\text{-ext1}_{\zeta+\rho}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H}],B,\mathsf{Ext}}(k) + 2\cdot\mathbf{Adv}^{\text{n-cr}_{\zeta+\rho}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H}],C}(k) \\ & +2\cdot\mathbf{Adv}^{\text{hcf}}_{\mathcal{F}|_\zeta,\mathcal{H},\mathbf{X},D}(k) + 2v\cdot\mathbf{Adv}^{\text{prg}}_{\mathcal{G},E}(k)\ .\end{aligned}$$

EFFICIENCY. The ciphertext length in the above instantiations is $3n + 3k$ where n is the length of the RSA modulus and k is the security parameter. The scheme has message length n. For example, if $n = 2048$ and $k = 128$ then the ciphertext length is 6528 bits. The time to run the encryption and decryption algorithms is basically that of standard RSA-OAEP.

Remark 6.3 We note that while the restriction to public-key independent messages is inherent for deterministic encryption, for randomized encryption it is not and we leave it as an interesting open problem to extend the result to public-key dependent messages. Additionally, the high-entropy requirement on messages can be avoided by assuming $\mathcal{G}$ and $\mathcal{H}$ are "universal computational extractors" (UCE) in the sense of [7], which follows from their results but we omit this for simplicity.

6.3 \$IND-CPA-KI result

We first show that t-clear RSA-OAEP is \$IND-CPA-KI for messages independent of the public key. Note that the prior result on full instantiability of t-clear RSA-OAEP by Boldyreva and Fischlin [22] also apply to public-key-independent messages. Intuitively, we require additional high min-entropy message sampler because the randomness for the distribution on which $\mathcal{H}$ is hardcore for $\mathcal{F}|_\zeta$ comes from message m, with coins r fixed. Again, we note that we could avoid the high min-entropy requirement by using the result in [7] that RSA-OAEP if RSA is one-way and $\mathcal{G}, \mathcal{H}$ are UCE. We prefer to stick with our more mild assumptions as our result is novel with a non-trivial proof and probably sufficient in practice. (The UCE result could be seen as a hedge.)

Theorem 6.4 Let μ, ζ, ρ be integer parameters. Let $\mathcal{F}$ be a family of one-way trapdoor permutations with domain $\{0,1\}^\mu$. Let $\mathscr{M}$ be a class of (ℓ, v)-entropic message samplers. Suppose $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ is

Games $G_1(k)$, $G_2(k)$	**Games** $G_3(k)$, $G_4(k)$
$b \leftarrow\$ \{0,1\}$; $param \leftarrow\$ A.\mathrm{pg}(1^k)$	$b \leftarrow\$ \{0,1\}$; $param \leftarrow\$ A.\mathrm{pg}(1^k)$
$K_G \leftarrow\$ \mathcal{K}_G(1^k)$; $K_H \leftarrow\$ \mathcal{K}_H(1^k)$	$K_G \leftarrow\$ \mathcal{K}_G(1^k)$; $K_H \leftarrow\$ \mathcal{K}_H(1^k)$
$(f, f^{-1}) \leftarrow\$ \mathsf{Kg}(1^k)$	$(f, f^{-1}) \leftarrow\$ \mathsf{Kg}(1^k)$
$pk \leftarrow (K_G, K_H, f)$	$pk \leftarrow (K_G, K_H, f)$
$\mathbf{m} \leftarrow\$ \mathcal{M}_b(1^k, param)$	$\mathbf{m} \leftarrow\$ \mathcal{M}_b(1^k, param)$
For $i = 1$ to $\lvert\mathbf{m}\rvert$ do	For $i = 1$ to $\lvert\mathbf{m}\rvert$ do
$r \leftarrow\$ \{0,1\}^\rho$; $x \leftarrow G(K_G, r)$	$r \leftarrow\$ \{0,1\}^\rho$
$s \leftarrow \mathbf{m}[i] \Vert 0^\zeta \oplus x$; $s_1 \leftarrow s\vert^\mu$; $s_2 \leftarrow s\vert_\zeta$	$x \leftarrow G(K_G, r)$; $x \leftarrow\$ \{0,1\}^{\mu+\zeta}$
$z \leftarrow H(K_H, s)$; $z \leftarrow\$ \{0,1\}^\rho$	$s \leftarrow \mathbf{m}[i] \Vert 0^\zeta \oplus x$; $s_1 \leftarrow s\vert^\mu$; $s_2 \leftarrow s\vert_\zeta$
$t \leftarrow r \oplus z$; $y \leftarrow f(s_1)$	$t \leftarrow\$ \{0,1\}^\rho$; $z \leftarrow r \oplus t$
$\mathbf{c}[i] \leftarrow (y, s_2, t)$	$y \leftarrow f(s_1)$; $\mathbf{c}[i] \leftarrow (y, s_2, t)$
$b' \leftarrow\$ A.\mathrm{g}(pk, \mathbf{c}, param)$	$b' \leftarrow\$ A.\mathrm{g}(pk, \mathbf{c}, param)$
Return $(b = b')$	Return $(b = b')$

Figure 34: **Games G_1–G_4 in the proof of Theorem 6.4.**

a pseudorandom generator and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ is a hardcore function for $\mathcal{F}|_\zeta$ on class $\mathscr{M}$. Then $\mathsf{OAEP}_{\mathsf{t\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|_{\zeta+\rho}]$ is \$IND-CPA-KI secure. In particular, for every $\mathcal{M}_0, \mathcal{M}_1 \in \mathscr{M}$ and any adversary A, there are adversaries B, C and distribution $\mathbf{X}(k) \in \mathscr{M}$ such that

$$\mathbf{Adv}^{\$\text{ind-cpa-ki}}_{\mathsf{OAEP}_{\mathsf{t\text{-}clear}}, A, \mathcal{M}_0, \mathcal{M}_1}(k) \le 2 \cdot \mathbf{Adv}^{\mathrm{hcf}}_{\mathcal{F}|_\zeta, \mathcal{H}, \mathbf{X}, B}(k) + 2v \cdot \mathbf{Adv}^{\mathrm{prg}}_{\mathcal{G}, C}(k) \ .$$

The running time of B is up to that of A. The running time of $\mathbf{X}(k)$ and C is the time to run A plus the running time of $\mathcal{M}$.

Proof: Consider games G_1–G_4 in Figure 34. Game G_1 corresponds to game \$IND-CPA-KI$^{A,\mathcal{M}_0,\mathcal{M}_1}_{\mathsf{OAEP}_{\mathsf{t\text{-}clear}}}$. We now explain the game chain. Game G_2 is identical to game G_1, except we are using completely random z in the encryption phase instead of using the hash value. Consider distribution $\mathbf{X}$ and adversary B in Figure 35. Note that $\mathbf{X}(k) \in \mathscr{M}$. Distribution $\mathbf{X}$ and adversary B collaborate to simulate game G_1. Adversary B returns 0 if adversary A can correctly guess simulated challenge bit b, and returns 1 otherwise. Then

$$\Pr[G_1(k) \Rightarrow 1] - \Pr[G_2(k) \Rightarrow 1] \le \mathbf{Adv}^{\mathrm{hcf}}_{\mathcal{F}|_\zeta, \mathcal{H}, \mathbf{X}, B}(k) \ .$$

In game G_3, we reorder the code of game G_2 producing z. The change is conservative, meaning that $\Pr[G_2(k) \Rightarrow 1] = \Pr[G_3(k) \Rightarrow 1]$. Next, game G_4 is identical to game G_3, except we are using completely random x instead of pseudorandom value $G(K_G, r)$ in the encryption phase. For $i \in [v]$, we define the adversary C_i as shown in Figure 36. Hence,

$$\Pr[G_3 \Rightarrow 1] - \Pr[G_4 \Rightarrow 1] \le \sum_{i=1}^{v} \mathbf{Adv}^{\mathrm{prg}}_{\mathcal{G}, C_i}(k) \ .$$

Assume there exists adversary C such that for all $i \in [v]$, we have $\mathbf{Adv}^{\mathrm{prg}}_{\mathcal{G}, C_i}(k) \le \mathbf{Adv}^{\mathrm{prg}}_{\mathcal{G}, C}(k)$. Note that $\Pr[G_4(k) \Rightarrow 1] = 1/2$, since the distribution of the ciphertexts is completely independent of bit b. Summing up,

$$\mathbf{Adv}^{\$\text{ind-cpa-ki}}_{\mathsf{OAEP}_{\mathsf{t\text{-}clear}}, A, \mathcal{M}_0, \mathcal{M}_1}(k) \le 2 \cdot \mathbf{Adv}^{\mathrm{hcf}}_{\mathcal{F}|_\zeta, \mathcal{H}, \mathbf{X}, B}(k) + 2v \cdot \mathbf{Adv}^{\mathrm{prg}}_{\mathcal{G}, C}(k) \ .$$

This completes the proof. ∎

6.4 PA0 and PA1 results

PA0 RESULT. We show PA0 of t-clear RSA-OAEP. As mentioned above, here we obtain a better result by using properties of the round functions of OAEP directly, rather than properties of the overall padding scheme.

Theorem 6.5 Let $\eta, \delta, \mu, \zeta, \rho$ be integer parameters. Let $\mathcal{F}$ be a family of one-way trapdoor permutations with domain $\{0,1\}^\mu$. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be function families. Let

Algorithm $\mathbf{X}(k)$	**Algorithm** $B(K_H, f, \mathbf{y}, \alpha, \mathbf{z})$
$b \leftarrow\!\!\$\ \{0,1\}$; $param \leftarrow\!\!\$\ A.\mathrm{pg}(1^k)$	$(\mathbf{r}, \mathbf{s}_2, K_G, b, param) \leftarrow \alpha$
$K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$	$pk \leftarrow (K_G, K_H, f)$
$\mathbf{m} \leftarrow\!\!\$\ \mathcal{M}_b(1^k, param)$	For $i = 1$ to $\lvert\mathbf{z}\rvert$ do
For $i = 1$ to $\lvert\mathbf{m}\rvert$ do	$\quad t \leftarrow \mathbf{r}[i] \oplus \mathbf{z}[i]$
$\quad \mathbf{r}[i] \leftarrow\!\!\$\ \{0,1\}^\rho$; $x \leftarrow G(K_G, \mathbf{r}[i])$	$\quad \mathbf{c}[i] \leftarrow (\mathbf{y}[i], \mathbf{s}_2[i], t)$
$\quad s \leftarrow \mathbf{m}[i] \Vert 0^\zeta \oplus x$; $\mathbf{s}_1[i] \leftarrow s\vert^\mu$; $\mathbf{s}_2[i] \leftarrow s\vert_\zeta$	$b' \leftarrow\!\!\$\ A.\mathrm{g}(pk, \mathbf{c}, param)$
$\alpha \leftarrow (\mathbf{r}, \mathbf{s}_2, K_G, b, param)$	Return $(b \neq b')$
Return $(\mathbf{s}_1, \alpha)$	

Figure 35: **Distribution X (left) and adversary B (right) in the proof of Theorem 6.4.**

Adversary $C_i(K_G, x)$
$b \leftarrow\!\!\$\ \{0,1\}$; $param \leftarrow\!\!\$\ A.\mathrm{pg}(1^k)$
$K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$; $(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$
$pk \leftarrow (K_G, K_H, f)$; $\mathbf{m} \leftarrow\!\!\$\ \mathcal{M}_b(1^k, param)$
For $j = 1$ to $\lvert\mathbf{m}\rvert$ do
$\quad$ If $j < i$ then $\mathbf{x}[j] \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$
$\quad$ If $j = i$ then $\mathbf{x}[j] \leftarrow\!\!\$\ x$
$\quad$ If $j > i$ then
$\quad\quad r \leftarrow\!\!\$\ \{0,1\}^\rho$; $\mathbf{x}[j] \leftarrow\!\!\$\ G(K_G, r)$
$\quad s \leftarrow \mathbf{m}[j] \Vert 0^\zeta \oplus \mathbf{x}[j]$; $\mathbf{s}_1[j] \leftarrow s\vert^\mu$; $\mathbf{s}_2[j] \leftarrow s\vert_\zeta$
$\quad \mathbf{y}[j] \leftarrow f(\mathbf{s}_1[j])$; $\boldsymbol{t}[j] \leftarrow\!\!\$\ \{0,1\}^\rho$
$\quad \mathbf{c}[j] \leftarrow (\mathbf{y}[j], \mathbf{s}_2[j], \boldsymbol{t}[j])$
$b' \leftarrow\!\!\$\ A.\mathrm{g}(pk, \mathbf{c}, param)$
Return $(b = b')$

Figure 36: **Adversary C_i in the proof of Theorem 6.4.**

$\eta = \lvert[\mathsf{Kg}(1^k)]\rvert + \lvert[\mathcal{K}_H(1^k)]\rvert$ and $\delta = \lvert[\mathsf{Kg}(1^k)]\rvert + \lvert[\mathcal{K}_G(1^k)]\rvert$. Suppose $\mathcal{G}$ is η-EXT0$_\zeta$ and NCR$_\zeta$, $\mathcal{H}$ is (δ,ζ)-EXT0 and CR. Then $\mathsf{OAEP}_{\mathsf{t\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}\vert_{\zeta+\rho}]$ is PA0 secure. In particular, for any adversary A, there exist adversaries A_G, A_H, B_G, B_H and an extractor Ext such that for all extractors $\mathsf{Ext}_G, \mathsf{Ext}_H$

$$\begin{aligned}\mathbf{Adv}^{\mathrm{pa0}}_{\mathsf{OAEP}_{\mathsf{t\text{-}clear}}, A, \mathsf{Ext}}(k) \;\leq\; & \mathbf{Adv}^{\eta\text{-ext0}_\zeta}_{\mathcal{G}, A_G, \mathsf{Ext}_G}(k) + \mathbf{Adv}^{(\delta,\zeta)\text{-ext0}}_{\mathcal{H}, A_H, \mathsf{Ext}_H}(k) \\ & + \mathbf{Adv}^{\mathrm{n\text{-}cr}_\zeta}_{\mathcal{G}, B_G}(k) + \mathbf{Adv}^{\mathrm{cr}}_{\mathcal{H}, B_H}(k) \ .\end{aligned}$$

The running time of A_G is about that of A and the running time of A_H is about that of A plus the time to run Ext_G. The running time of B_G is the time to run A and Ext_G. The running time of B_H is the running time of A, Ext_H and Ext_G. The running time of Ext is the time to run Ext_G and Ext_H.

Proof: Let w be the randomness of adversary A in the game PA0. Let K_G be the key for the function family $\mathcal{G}$, K_H be the key for the function family $\mathcal{H}$ and f be the evaluation key for the trapdoor permutation family $\mathcal{F}$ in the game PA0. We define EXT0 adversaries A_G, A_H with randomness w in Figure 37. Let $v = (K_H, f)$ be the key independent auxiliary input to A_G and $u = (K_G, f)$ be the key independent auxiliary input to A_H. Note that auxiliary input v and u are independent of key K_G and K_H, respectively. Let Ext_G and Ext_H be the corresponding extractor for A_G and A_H, respectively. We define PA0 extractor Ext as shown in Figure 38.

Note that for decryption query c that A makes, if ciphertext c is not valid then extractor Ext outputs $\perp$. Thus, adversary A does not gain any information about b by making an invalid decryption query. Hence, we assume wlog that adversary A queries only valid ciphertexts. Let c be the decryption query that A makes and r_c and s_c be the corresponding middle values in the computation of the ciphertext c. Let r be the output of Ext_G and s_1 be the output of Ext_H. Wlog, we can assume when Ext_G and Ext_H output a non-empty string, they were successful in finding the preimages. Let W_1 be the event that r is a non-empty string and $r \neq r_c$ and W_2 be the event that s is a non-empty string and $s_1 \neq s_c\vert^\mu$. Let $W = W_1 \vee W_2$. Let S be the event that game $\mathrm{PA0}^{A, \mathsf{Ext}}_{\mathsf{OAEP}_{\mathsf{t\text{-}clear}}}(k)$ outputs

Adversary $A_H(K_H, u; w)$	**Adversary** $A_G(K_G, v; w)$
$\mathsf{out} \leftarrow \bot$; $(K_G, f) \leftarrow u$	$\mathsf{out} \leftarrow \bot$; $(K_H, f) \leftarrow v$
$pk \leftarrow (K_G, K_H, f)$	$pk \leftarrow (K_G, K_H, f)$
Run $A^{\mathcal{D}\textsc{Sim}(\cdot)}(pk; w)$	Run $A^{\mathcal{D}\textsc{Sim}(\cdot)}(pk; w)$
Return (s_2, out)	Return $(\bot, \mathsf{out})$
Procedure $\mathcal{D}\textsc{Sim}(c)$	**Procedure** $\mathcal{D}\textsc{Sim}(c)$
$(y, s_2, t) \leftarrow c$; $v \leftarrow (K_H, f)$	$(y, s_2, t) \leftarrow c$
$r \leftarrow \mathsf{Ext}_G(K_G, v, \bot, s_2; w)$	$\mathsf{out} \leftarrow s_2$
$\mathsf{out} \leftarrow r \oplus t$	Halt A
Halt A	

Figure 37: **Adversaries A_H, A_G in the proof of Theorem 6.5.**

Algorithm $\mathsf{Ext}(pk, w, c)$
$(y, s_2, t) \leftarrow c$; $(K_G, K_H, f) \leftarrow pk$
$v \leftarrow (K_H, f)$; $u \leftarrow (K_G, f)$
$r \leftarrow \mathsf{Ext}_G(K_G, v, \bot, s_2; w)$; $z \leftarrow r \oplus t$
$s_1 \leftarrow \mathsf{Ext}_H(K_H, u, s_2, z; w)$; $s \leftarrow s_1 \| s_2$
$m^* \leftarrow s \oplus G(K_G, r)$; $m \leftarrow m^*|^\mu$
If $m^*|_\zeta \neq 0^\zeta$ then return $\bot$
If $c \neq \mathsf{Enc}(pk, m; r)$ then return $\bot$
Return m

Figure 38: **PA0 extractor Ext in the proof of Theorem 6.5.**

1. Note that all of the following probabilities are over the choice of public key pk and randomness w. Then

$$\mathbf{Adv}^{\mathrm{pa0}}_{\mathsf{OAEP}_{\mathsf{t\text{-}clear}}, A, \mathsf{Ext}}(k) = 2 \cdot \left(\Pr\left[S \wedge W \right] + \Pr\left[S \wedge \overline{W} \right] \right) - 1 \ .$$

Now consider near collision resistance adversary B_G in Figure 39 and collision resistance adversary B_H in Figure 40. If event W happens, then either B_G or B_H finds a collision. Thus, we have $\Pr[W] \leq \mathbf{Adv}^{\mathrm{n\text{-}cr}_\zeta}_{\mathcal{G}, B_G}(k) + \mathbf{Adv}^{\mathrm{cr}}_{\mathcal{H}, B_H}(k)$. Then

$$\Pr\left[S \wedge W \right] \leq \mathbf{Adv}^{\mathrm{n\text{-}cr}_\zeta}_{\mathcal{G}, B_G}(k) + \mathbf{Adv}^{\mathrm{cr}}_{\mathcal{H}, B_H}(k) \ .$$

Let E be the event such that for decryption query c adversary A makes, the outputs of decryption algorithm Dec and extractor Ext are equal. Then,

$$\Pr\left[S \wedge \overline{W} \right] = \Pr\left[S \wedge \overline{W} \wedge E \right] + \Pr\left[S \wedge \overline{W} \wedge \overline{E} \right] \ .$$

Note that W and E are mutually exclusive and when event W happens the output of Ext is incorrect. Thus, we have $\Pr\left[S \wedge \overline{W} \wedge E \right] = \Pr[S \wedge E]$. Moreover, we have $\Pr[S \mid E] = 1/2$ since for the query made by A the outputs of Dec and Ext are equal. Hence,

$$\Pr[S \wedge E] = \frac{1}{2} \cdot \Pr[E] \ .$$

Consider adversaries A_G, A_H in Figure 37. We know adversary A always makes valid decryption query. Thus, when event $\overline{E}$ happens extractor Ext outputs $\bot$. This implies that either $r \neq r_c$ where r is the output of Ext_G or $s_1 \neq s_c|^\mu$ where s_1 is the output of Ext_H. We also know when event $\overline{W}$ happens, extractor Ext_G either outputs $\bot$ or r_c, and extractor Ext_H outputs either $\bot$ or $s_c|^\mu$. Therefore when events $\overline{E}$ and $\overline{W}$ happen, either Ext_G or Ext_H fails. Thus,

$$\Pr\left[\overline{W} \wedge \overline{E} \right] \leq \mathbf{Adv}^{\eta\text{-ext0}_\zeta}_{\mathcal{G}, A_G, \mathsf{Ext}_G}(k) + \mathbf{Adv}^{(\delta,\zeta)\text{-ext0}}_{\mathcal{H}, A_H, \mathsf{Ext}_H}(k) \ .$$

On the other hand, we know that E and W mutually exclusive. Hence, we get $\Pr[E] = \Pr[W \vee E] - \Pr[W]$.

Adversary $B_G(K_G)$	**Procedure** $\mathcal{D}\text{SIM}(c)$
$\mathsf{out}_1 \leftarrow \varepsilon$; $\mathsf{out}_2 \leftarrow \varepsilon$	$(y, s_2, t) \leftarrow c$; $s \leftarrow f^{-1}(y) \| s_2$
$w \leftarrow\!\!\$\ \mathsf{Coins}(k)$	$\mathsf{out}_1 \leftarrow \mathsf{Ext}_G(K_G, v, \perp, s_2; w)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	$\mathsf{out}_2 \leftarrow t \oplus H(K_H, s)$
$K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$	Halt A
$v \leftarrow (K_H, f)$	
$pk \leftarrow (K_G, K_H, f)$	
Run $A^{\mathcal{D}\text{SIM}(\cdot)}(pk; w)$	
Return $(\mathsf{out}_1, \mathsf{out}_2)$	

Figure 39: **CR adversary B_G in the proof of Theorem 6.5.**

Adversary $B_H(K_H)$	**Procedure** $\mathcal{D}\text{SIM}(c)$
$\mathsf{out}_1 \leftarrow \varepsilon$; $\mathsf{out}_2 \leftarrow \varepsilon$	$(y, s_2, t) \leftarrow c$; $\mathsf{out}_1 \leftarrow f^{-1}(y) \| s_2$
$w \leftarrow\!\!\$\ \mathsf{Coins}(k)$	$r \leftarrow \mathsf{Ext}_G(K_G, v, \perp, s_2; w)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	$z \leftarrow t \oplus r$
$K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$	$s_1 \leftarrow \mathsf{Ext}_H(K_H, u, s_2, z; w)$
$v \leftarrow (K_H, f)$	$\mathsf{out}_2 \leftarrow s_1 \| s_2$
$u \leftarrow (K_G, f)$	Halt A
$pk \leftarrow (K_G, K_H, f)$	
Run $A^{\mathcal{D}\text{SIM}(\cdot)}(pk; w)$	
Return $(\mathsf{out}_1, \mathsf{out}_2)$	

Figure 40: **CR adversary B_H in the proof of Theorem 6.5.**

Summing up,

$$\begin{aligned}\mathbf{Adv}^{\text{pa0}}_{\mathsf{OAEP}_{\mathsf{t\text{-}clear}},A,\mathsf{Ext}}(k) &\leq 2 \cdot \Pr[W] + (\Pr[W \vee E] - \Pr[W]) + 2 \cdot \Pr\left[\overline{W} \wedge \overline{E}\right] - 1 \\ &\leq \mathbf{Adv}^{\eta\text{-ext0}_\zeta}_{\mathcal{G},A_G,\mathsf{Ext}_G}(k) + \mathbf{Adv}^{(\delta,\zeta)\text{-ext0}}_{\mathcal{H},A_H,\mathsf{Ext}_H}(k) + \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G},B_G}(k) + \mathbf{Adv}^{\text{cr}}_{\mathcal{H},B_H}(k) \ .\end{aligned}$$

This completes the proof ∎

PA1 RESULT. We now show that t-clear RSA-OAEP "inherits" the extractability of the underlying padding transform, in the form of PA1 and EXT1, as long as the latter is also near-collision resistant. Here we state the result for an abstract padding scheme rather than specifically for OAEP. Interestingly, this approach does not seem to work for PA2 and EXT2. We leave PA2 of the encryption scheme as an open problem.

Theorem 6.6 Let δ, μ, ζ, ρ be integer parameters. Let $\mathcal{F}$ be a family of one-way trapdoor permutations with domain $\{0,1\}^\mu$ and $\delta = |[\mathsf{Kg}(1^k)]|$. Let PAD be a padding transform from domain $\{0,1\}^{\mu+\rho}$ to range $\{0,1\}^{\mu+\zeta+\rho}$. Suppose PAD is $\text{NCR}_{\zeta+\rho}$ and $\delta\text{-EXT1}_{\zeta+\rho}$. Then the padding-based encryption scheme $\mathsf{PAD}[\mathcal{F}|_{\zeta+\rho}]$ is PA1 secure. In particular, for any PA1 adversary A that makes q queries, there exist an EXT1 adversary A_{PAD} that makes q queries, an $\text{NCR}_{\zeta+\rho}$ adversary B and an extractor Ext such that for all extractors $\mathsf{Ext}_{\mathsf{PAD}}$

$$\mathbf{Adv}^{\text{pa1}}_{\mathsf{PAD}[\mathcal{F}|_{\zeta+\rho}],A,\mathsf{Ext}}(k) \leq \mathbf{Adv}^{\delta\text{-ext1}_{\zeta+\rho}}_{\mathsf{PAD},A_{\mathsf{PAD}},\mathsf{Ext}_{\mathsf{PAD}}}(k) + \mathbf{Adv}^{\text{n-cr}_{\zeta+\rho}}_{\mathsf{PAD},B}(k) \ .$$

The running time of A_{PAD} is that of A. The running time of B is about the time to run A, $\mathsf{Ext}_{\mathsf{PAD}}$ and Ext. The running time of Ext is about that of $\mathsf{Ext}_{\mathsf{PAD}}$.

Proof: Let w be the randomness of PA1 adversary A. We define EXT1 adversary A_{PAD} with randomness w in Figure 41. Note that auxiliary input f is independent of the key π. Let $\mathsf{Ext}_{\mathsf{PAD}}$ be the corresponding extractor for A_{PAD}. We define PA1 extractor Ext as shown in Figure 42.

Note that for any decryption query c adversary A makes, if c is not valid then extractor Ext outputs $\perp$. Thus, A does not gain any information about b by making invalid decryption queries. Hence, we assume wlog that adversary A only makes queries for valid ciphertext c. Assume A makes q extract queries. Let c_i be the i-th query A makes to the extract oracle and v_i be the output of extractor $\mathsf{Ext}_{\mathsf{PAD}}$ on input $c_i|_{\zeta+\rho}$. Let W be the

Adversary $A_{\mathsf{PAD}}^{\mathcal{O}_{\mathsf{PAD}}(\cdot)}(\pi, f; w)$
$pk \leftarrow (\pi, f)$
Run $A^{\mathcal{D}\textsc{Sim}(\cdot)}(pk; w)$

Procedure $\mathcal{D}\textsc{Sim}(c)$
$v \leftarrow \mathcal{O}_{\mathsf{PAD}}(\bot, c|_{\zeta+\rho})$
Return $v|^{\mu}$

Figure 41: **EXT1 adversary A_{PAD} in the proof of Theorem 6.6.**

Algorithm $\mathsf{Ext}(state, c)$
$(pk, w, st) \leftarrow state$; $(\pi, f) \leftarrow pk$
$(v, st) \leftarrow \mathsf{Ext}_{\mathsf{PAD}}(st, \pi, f, \bot, c|_{\zeta+\rho}; w)$
$state \leftarrow (pk, w, st)$
$m \leftarrow v|^{\mu}$; $r \leftarrow v|_{\rho}$
If $c \neq \mathsf{Enc}(pk, m; r)$ then return $(\bot, state)$
Return $(m, state)$

Figure 42: **PA1 extractor Ext in the proof of Theorem 6.6.**

event where there exists a valid ciphertext c_i such that v_i is non-empty and extractor Ext outputs $\bot$ on input c_i. Let S be the event that game $\mathrm{PA1}^{A,\mathsf{Ext}}_{\mathsf{PAD}[\mathcal{F}|_{\zeta+\rho}]}(k)$ outputs 1. Note that all of the following probabilities are over the choice of public key pk and randomness w. Then,

$$\mathbf{Adv}^{\mathrm{pa1}}_{\mathsf{PAD}[\mathcal{F}|_{\zeta+\rho}],A,\mathsf{Ext}}(k) = 2 \cdot \big(\Pr[\, S \wedge W \,] + \Pr\big[\, S \wedge \overline{W} \,\big]\big) - 1 \ .$$

Note that extractor $\mathsf{Ext}_{\mathsf{PAD}}$ either outputs the correct value or $\bot$. Now consider near-collision resistance adversary B in Figure 43. If event W happens, then B finds a collision. Thus, we have $\Pr[\, W \,] \leq \mathbf{Adv}^{\text{n-cr}_{\zeta+\rho}}_{\mathsf{PAD},B}(k)$. Then

$$\Pr[\, S \wedge W \,] \leq \mathbf{Adv}^{\text{n-cr}_{\zeta+\rho}}_{\mathsf{PAD},B}(k) \ .$$

Let E be the event such that for each decryption query c_i that adversary A makes, the output of decryption algorithm Dec and extractor Ext are equal. Then,

$$\Pr\big[\, S \wedge \overline{W} \,\big] = \Pr\big[\, S \wedge \overline{W} \wedge E \,\big] + \Pr\big[\, S \wedge \overline{W} \wedge \overline{E} \,\big] \ .$$

Note that W and E are mutually exclusive since when event W happens the output of the extractor Ext would be incorrect for at least one of the extract queries. Thus, we have $\Pr\big[\, S \wedge \overline{W} \wedge E \,\big] = \Pr[\, S \wedge E \,]$. Moreover, we have $\Pr[\, S \mid E \,] = 1/2$, since for all queries made by A the output of decryption oracle Dec and extractor Ext are equal. Hence,

$$\Pr[\, S \wedge E \,] = \frac{1}{2} \cdot \Pr[\, E \,] \ .$$

Consider the adversary A_{PAD} in Figure 41. We know adversary A always makes valid decryption query. Thus, when event $\overline{E}$ happens extractor Ext outputs $\bot$ for at least one of the extract queries. Moreover, if $\overline{W}$ happens then for all extract queries either $\mathsf{Ext}_{\mathsf{PAD}}$ outputs $\bot$ or Ext outputs non-empty string. Therefore, when $\overline{E}$ and $\overline{W}$ happen, extractor $\mathsf{Ext}_{\mathsf{PAD}}$ fails and outputs $\bot$ for at least one of the extract queries. Thus,

$$\Pr\big[\, \overline{W} \wedge \overline{E} \,\big] \leq \mathbf{Adv}^{\delta\text{-ext1}_{\zeta+\rho}}_{\mathsf{PAD},A_{\mathsf{PAD}},\mathsf{Ext}_{\mathsf{PAD}}}(k) \ .$$

On the other hand, we know that E and W mutually exclusive. Hence, we get $\Pr[\, E \,] = \Pr[\, W \vee E \,] - \Pr[\, W \,]$. Summing up,

$$\mathbf{Adv}^{\mathrm{pa1}}_{\mathsf{PAD}[\mathcal{F}|_{\zeta+\rho}],A,\mathsf{Ext}}(k) \leq \mathbf{Adv}^{\delta\text{-ext1}_{\zeta+\rho}}_{\mathsf{PAD},A_{\mathsf{PAD}},\mathsf{Ext}_{\mathsf{PAD}}}(k) + \mathbf{Adv}^{\text{n-cr}_{\zeta+\rho}}_{\mathsf{PAD},B}(k) \ .$$

This completes the proof. ∎

Adversary $B(\pi, \hat{\pi})$	**Procedure** $\mathcal{D}\text{SIM}(c)$
$\mathsf{out}_1 \leftarrow \varepsilon$; $\mathsf{out}_2 \leftarrow \varepsilon$	$y_2 \leftarrow c\|_{\zeta+\rho}$; $y_1 \leftarrow f^{-1}(c\|^{\mu})$
$w \leftarrow\!\!\$\ \mathsf{Coins}(k)$; $b \leftarrow\!\!\$\ \{0,1\}$	$y \leftarrow y_1 \| y_2$; $\mathsf{out}_1 \leftarrow \hat{\pi}(y)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	$(\mathsf{out}_2, st) \leftarrow \mathsf{Ext}_{\mathsf{PAD}}(st, \pi, f, \bot, c\|_{\zeta+\rho}; w)$
$pk \leftarrow (\pi, f)$; $sk \leftarrow (\hat{\pi}, f^{-1})$	If $\big(\pi(\mathsf{out}_1)\|_{\zeta+\rho} = \pi(\mathsf{out}_2)\|_{\zeta+\rho} \wedge \mathsf{out}_1 \neq \mathsf{out}_2\big)$
$st \leftarrow (\pi, w)$; $state \leftarrow (pk, w)$	Halt A
Run $A^{\mathcal{D}\text{SIM}(\cdot)}(pk; w)$	$m_0 \leftarrow \mathsf{Dec}(sk, c)$
Return $(\mathsf{out}_1, \mathsf{out}_2)$	$(m_1, state) \leftarrow\!\!\$\ \mathsf{Ext}(state, c)$
	Return m_b

Figure 43: **NCR adversary B in the proof of Theorem 6.6.**

7 Full Instantiation Results for s-Clear RSA-OAEP

In this section, we give full instantiation results for s-clear RSA-OAEP. Note that we are the first to consider this variant. We show that s-clear is IND-CCA2 if $\mathcal{G}$ is a pseudorandom generator, near-collision resistant, and "many-times" extractable with dependent auxiliary information, $\mathcal{H}$ is collision-resistant, and $\mathcal{F}$ meets novel "XOR-nonmalleability" and "XOR-indistinguishability" notions that seem plausible for RSA. Also note that we avoid the several impossibility results here. First, we avoid the impossibility result of [69] by using XOR-non-malleability of $\mathcal{F}$. Second, we avoid the impossibility result of [16] since the dependent auxiliary information is bounded.

7.1 XOR Assumptions on Trapdoor Permutations and RSA

Here, we give classes of novel assumptions on RSA (and trapdoor permutations in general), which are stronger than one-wayness and needed for RSA-OAEP s-clear.

XOR-IND. Our first class of assumptions speaks to the fact that addition or XOR operations "break up" the multiplicative structure of RSA. Indeed, in a related context of arithmetic progressions on $\mathbb{Z}_N$ we have seen formal evidence of this [60, 71]. It is interesting for future work to give formal evidence in our case as well. Let $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a trapdoor permutation family with domain TDom. Let $\mathcal{G} : \mathcal{K}_G \times \mathsf{TDom} \to \mathsf{GRng}$ be a function family. For ATK $\in \{\text{IND0}, \text{IND1}, \text{IND2}\}$, we associate the experiment in Figure 44, for every $k \in \mathbb{N}$. Define the xor-atk advantage of A against $\mathcal{F}$ with the hint function family $\mathcal{G}$

$$\mathbf{Adv}^{\text{xor-atk}}_{\mathcal{F},\mathcal{G},A}(k) = 2 \cdot \Pr\left[\,\text{XOR-ATK}^{A}_{\mathcal{F},\mathcal{G}}(k) \Rightarrow 1\,\right] - 1 \ .$$

If atk = ind0, then $\mathcal{O} = \varepsilon$. We say that $\mathcal{F}$ is XOR-IND0 with respect to hint function family $\mathcal{G}$ if for every PPT attacker A, $\mathbf{Adv}^{\text{xor-ind0}}_{\mathcal{F},\mathcal{G},A}(k)$ is negligible in k. Similarly, if atk = ind1, then $\mathcal{O} = \mathcal{C}$, where $\mathcal{C}$ is a relation checker oracle that on input y_1, y_2 and ω work as follows:

$$\mathcal{C}(y_1, y_2, \omega) = \begin{cases} 1 & \text{if } \ \omega = f^{-1}(y_1) \oplus f^{-1}(y_2) \\ 0 & \text{otherwise} \end{cases} \ .$$

Similarly, if atk = ind2$_\ell$, then $\mathcal{O} = \mathcal{V}_\ell$, where $\mathcal{V}_\ell$ is an ℓ-bit image verifier oracle that on input y works as follows:

$$\mathcal{V}_\ell(y) = \begin{cases} 1 & \text{if } \ \exists x \colon y = G(K_G, x)|_\ell \\ 0 & \text{otherwise} \end{cases} \ .$$

Note that adversary A is not allowed to query for the challenge to the image verifier oracle $\mathcal{V}_\ell$. We say that $\mathcal{F}$ is XOR-IND1 (resp. XOR-IND2$_\ell$) with respect to hint function family $\mathcal{G}$ if for every PPT attacker A, $\mathbf{Adv}^{\text{xor-ind1}}_{\mathcal{F},\mathcal{G},A}(k)$ (resp. $\mathbf{Adv}^{\text{xor-ind2}_\ell}_{\mathcal{F},\mathcal{G},A}(k)$) is negligible in k.

Observe that the hint is *crucial*, as otherwise the assumption would trivially hold. In our results, $\mathcal{G}$ is a PRG. In this case, we show that $\mathcal{G}$ is also a HCF function for $\mathcal{F}$. In other words, the assumption in our use-case can be viewed an extension of the classical notion of HCF — $\mathcal{G}$ is "robust" not in the sense of [47], but in the sense that the view of the adversary is also indistinguishable given $\mathcal{F}$ applied to either the real input or *related one*. Note that not all hardcore functions have this property, even when $\mathcal{F}$ is partial one-way. For example, consider

Game XOR-ATK$^{A}_{\mathcal{F},\mathcal{G}}(k)$
$b \leftarrow\!\!\$\ \{0,1\}\ ;\ (f, f^{-1}) \leftarrow \mathsf{Kg}(1^k)$
$K_G \leftarrow \mathcal{K}_G(1^k)\ ;\ x \leftarrow\!\!\$\ \mathsf{TDom}(k)$
$(state, z) \leftarrow\!\!\$\ A_1(f, K_G, G(K_G, x))$
$y_0 \leftarrow f(x)\ ;\ y_1 \leftarrow f(x \oplus z)$
$b' \leftarrow\!\!\$\ A_2^{\mathcal{O}}(state, y_b)$
Return $(b = b')$

Figure 44: **Games to define XOR-ATK security.**

Games $G_1(k)$, $G_2(k)$	**Games** $G_3(k)$
$b \leftarrow\!\!\$\ \{0,1\}\ ;\ K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$	$b \leftarrow\!\!\$\ \{0,1\}\ ;\ K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$
$(f, f^{-1}) \leftarrow \mathsf{Kg}(1^k)$	$(f, f^{-1}) \leftarrow \mathsf{Kg}(1^k)$
$x \leftarrow\!\!\$\ \mathsf{TDom}(k)$	$x \leftarrow\!\!\$\ \mathsf{TDom}(k)$
$g_0 \leftarrow G(K_G, x)\ ;\ g_1 \leftarrow\!\!\$\ \mathsf{GRng}(k)$	$g_0 \leftarrow\!\!\$\ \mathsf{GRng}(k)\ ;\ g_1 \leftarrow\!\!\$\ \mathsf{GRng}(k)$
$y \leftarrow f(x)\ ;\quad y \leftarrow\!\!\$\ \mathsf{TDom}(k)$	$y \leftarrow\!\!\$\ \mathsf{TDom}(k)$
$b' \leftarrow\!\!\$\ A(K_G, f, y, g_b)$	$b' \leftarrow\!\!\$\ A(K_G, f, y, g_b)$
Return $(b = b')$	Return $(b = b')$

Figure 45: **Games G_1–G_3 in the proof of Theorem 7.1.**

a hardcore function $\mathcal{G}$ that reveals first bit of its input x. Then if a partial one-way function $\mathcal{F}$ also reveals the first bit of x, XOR-indistinguishability clearly does not hold.

Theorem 7.1 Let $\mathcal{F}$ be a family of one-way trapdoor permutations with domain TDom. Suppose $\mathcal{G} : \mathcal{K}_G \times \mathsf{TDom} \to \mathsf{GRng}$ is a pseudorandom generator and $\mathcal{F}$ is XOR-IND0 with respect to hint function family $\mathcal{G}$. Then $\mathcal{G}$ is a hardcore function for $\mathcal{F}$ on the uniform distribution. In particular, for any adversary A, there are adversaries B, C such that

$$\mathbf{Adv}^{\mathrm{hcf}}_{\mathcal{F},\mathcal{G},U,A}(k) \le 2 \cdot \mathbf{Adv}^{\text{xor-ind0}}_{\mathcal{F},\mathcal{G},B}(k) + 2 \cdot \mathbf{Adv}^{\mathrm{prg}}_{\mathcal{G},C}(k)\ .$$

The running time of B and C are about that of A.

Proof: Consider games G_1–G_3 in Figure 45. Game G_1 corresponds to game HCF-DIST$^{A,U}_{\mathcal{F},\mathcal{G}}(k)$. We now explain the game chain. Game G_2 is identical to game G_1, except we are using completely random y instead of using the pseudorandom value $f(x)$. Consider adversary B as shown in Figure 46. Note that adversary B simulate games G_1, G_2 with respect to its inputs. It returns 1 if adversary A can correctly guess the simulated challenge bit b, and returns 0 otherwise. Hence,

$$\Pr[G_1 \Rightarrow 1] - \Pr[G_2 \Rightarrow 1] \le \mathbf{Adv}^{\text{xor-ind0}}_{\mathcal{F},\mathcal{G},B}(k)\ .$$

Next, game G_3 is identical to game G_2, except we are using completely random g_0 instead of using the pseudorandom value $G(K_G, x)$. Consider adversary C as shown in Figure 46. Note that adversary C simulate games G_2, G_3 with respect to its inputs. It returns 1 if adversary A can correctly guess the simulated challenge bit b, and returns 0 otherwise. Hence,

$$\Pr[G_2 \Rightarrow 1] - \Pr[G_3 \Rightarrow 1] \le \mathbf{Adv}^{\mathrm{prg}}_{\mathcal{G},C}(k)\ .$$

Note that $\Pr[G_3(k) \Rightarrow 1] = 1/2$, since y, g_0 and g_1 are uniformly random. Summing up,

$$\mathbf{Adv}^{\mathrm{hcf}}_{\mathcal{F},\mathcal{G},U,A}(k) \le 2 \cdot \mathbf{Adv}^{\text{xor-ind0}}_{\mathcal{F},\mathcal{G},B}(k) + 2 \cdot \mathbf{Adv}^{\mathrm{prg}}_{\mathcal{G},C}(k)\ .$$

This completes the proof. ∎

XOR-NM0. Our second class of assumptions speak to the fact that RSA is non-malleable wrt. XOR. Intuitively, if RSA was XOR malleable, then since is multiplicatively homomorphic it would be (something like) fully homomorphic, which is unlikely. (Although we do not claim the exact formulation of our definitions imply a formal

Algorithm $B_1(f, K_G, G(K_G, x))$	**Algorithm** $C(K_G, g_0)$
$state \leftarrow (f, K_G, G(K_G, x))$	$b \leftarrow\!\!\$\ \{0,1\}$
$z \leftarrow\!\!\$\ \mathsf{TDom}(k)$	$(f, f^{-1}) \leftarrow \mathsf{Kg}(1^k)$
Return $(state, z)$	$g_1 \leftarrow\!\!\$\ \mathsf{GRng}(k)$
Algorithm $B_2(state, y)$	$y \leftarrow\!\!\$\ \mathsf{TDom}(k)$
$b \leftarrow\!\!\$\ \{0,1\}$	$b' \leftarrow\!\!\$\ A(K_G, f, y, g_b)$
$(f, K_G, G(K_G, x)) \leftarrow state$	Return $(b = b')$
$g_0 \leftarrow G(K_G, x)$	
$g_1 \leftarrow\!\!\$\ \mathsf{GRng}(k)$	
$b' \leftarrow\!\!\$\ A(K_G, f, y, g_b)$	
Return $(b = b')$	

Figure 46: **Adversary B (left) and adversary C (right) in the proof of Theorem 7.1.**

Game XOR-NM0$^A_{\mathcal{F}}(k)$	**Game** XOR-NM1$^A_{\mathcal{F},\mathcal{G}}(k)$
$(f, f^{-1}) \leftarrow \mathsf{Kg}(1^k)$	$(f, f^{-1}) \leftarrow \mathsf{Kg}(1^k)\ ;\ K_G \leftarrow \mathcal{K}_G(1^k)$
$x \leftarrow\!\!\$\ \mathsf{TDom}(k)$	$x \leftarrow\!\!\$\ \mathsf{TDom}(k)\ ;\ z \leftarrow G(K_G, x)$
$(\omega, y') \leftarrow\!\!\$\ A(f, f(x))$	$(\alpha, state) \leftarrow\!\!\$\ A_1(f, K_G, z)$
$x' \leftarrow f^{-1}(y')$	$(\omega, y') \leftarrow\!\!\$\ A_2(state, f(x \oplus \alpha))$
If $(\omega = x \oplus x') \wedge (\omega \neq 0)$	$x' \leftarrow f^{-1}(y')$
Return 1	If $(\omega \oplus \alpha = x \oplus x') \wedge (\omega \neq 0)$
Else return 0	Return 1
	Else return 0

Figure 47: **Games to define XOR-NM security.**

definition of fully homomorphic.) A similar argument was made by Hofheinz for a non-malleability assumption on the Paillier trapdoor permutation (which is additively homomorphic) wrt. multiplication [Assumption 4.2][52].

Let $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a trapdoor permutation family with domain TDom. To adversary A, we associate the experiment in Figure 47 for every $k \in \mathbb{N}$. We say that $\mathcal{F}$ is XOR-NM0 if for every PPT attacker A,

$$\mathbf{Adv}^{\text{xor-nm0}}_{\mathcal{F},A}(k) = \Pr\left[\,\text{XOR-NM0}^A_{\mathcal{F}}(k) \Rightarrow 1\,\right] \ .$$

is negligible in k.

XOR-NM1. Let $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a trapdoor permutation family with domain TDom. Let $\mathcal{G} : \mathcal{K}_G \times \mathsf{TDom} \to \mathsf{GRng}$ be a hash function family. To adversary A, we associate the experiment in Figure 47 for every $k \in \mathbb{N}$. We say that $\mathcal{F}$ is XOR-NM1 with respect to $\mathcal{G}$ if for every PPT adversary A,

$$\mathbf{Adv}^{\text{xor-nm1}}_{\mathcal{F},\mathcal{G},A}(k) = \Pr\left[\,\text{XOR-NM1}^A_{\mathcal{F},\mathcal{G}}(k) \Rightarrow 1\,\right] \ .$$

is negligible in k.

Relations between definitions. Interestingly, we show XOR-NM0 and XOR-IND1 together imply XOR-NM1.

Theorem 7.2 Let $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a trapdoor permutation family with domain TDom. Let $\mathcal{G} : \mathcal{K}_G \times \mathsf{TDom} \to \mathsf{GRng}$ be a function family. Suppose $\mathcal{F}$ is XOR-NM0 and XOR-IND1 with respect to $\mathcal{G}$. Then, $\mathcal{F}$ is XOR-NM1 with respect to $\mathcal{G}$. In particular, for any adversary A, there are adversaries B, C such that

$$\mathbf{Adv}^{\text{xor-nm1}}_{\mathcal{F},\mathcal{G},A}(k) \leq \mathbf{Adv}^{\text{xor-nm0}}_{\mathcal{F},\mathcal{G},B}(k) + 2 \cdot \mathbf{Adv}^{\text{xor-ind1}}_{\mathcal{F},\mathcal{G},C}(k) \ .$$

The running time of B and C are about that of A.

Proof: Consider games G_1–G_4 in Figure 48. Game G_1 corresponds to the game XOR-NM1$^A_{\mathcal{F},\mathcal{G}}(k)$. We now explain the game chain. Game G_2 is identical to game G_1, except instead of giving $f(x \oplus z)$ as an input to the adversary A_2 we are using the value $f(x)$. Consider adversary C as shown in Figure 49. Note that adversary C

Games $G_1(k)$, $G_2(k)$	**Games** $G_3(k)$, $G_4(k)$
$(f, f^{-1}) \leftarrow \mathsf{Kg}(1^k)$	$(f, f^{-1}) \leftarrow \mathsf{Kg}(1^k)$; $K_G \leftarrow \mathcal{K}_G(1^k)$
$K_G \leftarrow \mathcal{K}_G(1^k)$; $x \leftarrow\!\!\$\ \mathsf{TDom}(k)$	$x \leftarrow\!\!\$\ \mathsf{TDom}(k)$; $x' \leftarrow\!\!\$\ \mathsf{TDom}(k)$
$(z, state) \leftarrow\!\!\$\ A_1(K_G, f, G(K_G, x))$	$w \leftarrow G(K_G, x)$; $w \leftarrow G(K_G, x')$
$y \leftarrow f(x \oplus z)$; $y \leftarrow f(x)$	$(\omega, y') \leftarrow\!\!\$\ A(K_G, f, f(x), w)$
$(\omega, y') \leftarrow\!\!\$\ A_2(state, y)$; $x' \leftarrow f^{-1}(y')$	$x' \leftarrow f^{-1}(y')$
$\mathsf{out} \leftarrow (\omega \oplus z = x \oplus x') \wedge (\omega \neq 0)$	Return $((\omega = x \oplus x') \wedge (\omega \neq 0))$
$\mathsf{out} \leftarrow (\omega = x \oplus x') \wedge (\omega \neq 0)$	
Return out	

Figure 48: **Games G_1–G_4 in the proof of Theorem 7.2.**

Algorithm $C_1(f, K_G, G(K_G, x))$	**Algorithm** $D_1(f, K_G, G(K_G, x))$
$(z, state) \leftarrow\!\!\$\ A_1(K_G, f, G(K_G, x))$	$z \leftarrow\!\!\$\ \mathsf{TDom}(k)$
Return $(z, state)$	$state \leftarrow (f, K_G, G(K_G, x), z)$
Algorithm $C_2^{\mathcal{C}(\cdot,\cdot,\cdot)}(state, y_b)$	Return $(z, state)$
$(\omega, y') \leftarrow\!\!\$\ A_2(state, y_b)$	**Algorithm** $D_2^{\mathcal{C}(\cdot,\cdot,\cdot)}(state, y_b)$
$b' \leftarrow \mathcal{C}(y', y_b, \omega)$	$(\omega, y') \leftarrow\!\!\$\ A(state, y_b)$
Return b'	$b' \leftarrow \mathcal{C}(y', y_b, \omega)$
	Return b'

Figure 49: **Adversaries C and D in the proof of Theorem 7.2.**

simulate games G_1, G_2 with respect to it's inputs. Hence,

$$\Pr[G_1 \Rightarrow 1] - \Pr[G_2 \Rightarrow 1] \leq \mathbf{Adv}^{\text{xor-ind1}}_{\mathcal{F},\mathcal{G},C}(k) \ .$$

In game G_3, we merge the adversaries A_1, A_2 of game G_2. The change is conservative, meaning that $\Pr[G_2(k) \Rightarrow 1] = \Pr[G_3(k) \Rightarrow 1]$. Game G_4 is identical to game G_3, except instead of giving $G(K_G, x)$ as an input to adversary A_2 we are using $G(K_G, x')$ for uniformly random x'. Consider adversary D as shown in Figure 49. Note that adversary D simulate games G_3, G_4 with respect to it's inputs. Hence,

$$\Pr[G_3 \Rightarrow 1] - \Pr[G_4 \Rightarrow 1] \leq \mathbf{Adv}^{\text{xor-ind1}}_{\mathcal{F},\mathcal{G},D}(k) \ .$$

Note that Game G_4 corresponds to the game XOR-NM0$^A_{\mathcal{F},\mathcal{G}}(k)$. Thus,

$$\mathbf{Adv}^{\text{xor-nm1}}_{\mathcal{F},\mathcal{G},A}(k) \leq \mathbf{Adv}^{\text{xor-nm0}}_{\mathcal{F},\mathcal{G},B}(k) + 2 \cdot \mathbf{Adv}^{\text{xor-ind1}}_{\mathcal{F},\mathcal{G},C}(k) \ .$$

This completes the proof. ∎

Discussion. We caution that these are new assumptions and must be treated with care, although they have some intuitive appeal as discussed where they are introduced. It would be interesting for future work to establish theoretical constructions meeting them or show that RSA meets them under more well-studied assumptions.

7.2 Main Results

First, we establish the security of s-clear RSA-OAEP in the RO model. Then, we show that it is IND-CCA1 and IND-CCA2 secure under respective suitable assumptions. As in Section 3 we actually prove corresponding notions of IND-CPA + PA, yielding stronger results. We refer to Appendix D for the proof in the random oracle model.

IND-CCA2 result in RO model. First, note that the partial one-wayness result of [46] does not apply to this variant, and in fact the negative result of [70] *does* apply, demonstrating that one-wayness of the trapdoor permutation is not enough for the scheme to achieve IND-CCA2 security *even in the RO model*. We show that XOR-nonmalleability is sufficient.

Theorem 7.3 Let μ, ζ, ρ be integer parameters. Let $\mathcal{F}$ be a XOR-NM0 family of one-way trapdoor permutations with domain $\{0,1\}^\rho$. Suppose $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ is a RO and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ is collision-resistant. Then $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|^{\mu+\zeta}]$ is IND-CCA2 secure in the random oracle model. In particular, for any adversary A, there are adversaries B, C such that

$$\mathbf{Adv}^{\text{ind-cca2}}_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}},A}(k) \leq \frac{2q}{2^\rho} + \frac{4p}{2^\zeta} + 2 \cdot \mathbf{Adv}^{\text{cr}}_{\mathcal{H},C}(k) + 4 \cdot \mathbf{Adv}^{\text{xor-nm0}}_{\mathcal{F},B}(k) \ .$$

where p is the number of decryption-oracle queries of A and q is the total number of random-oracle queries A and $\mathcal{M}$ make. Adversary B and C make at most q random-oracle queries. The running time of B and C are about that of A.

IND-CCA1 RESULT. To prove IND-CCA1, we use EXT1 and near-collision resistance of the overall OAEP padding scheme (which follows from assumptions on the round functions as per Section 5), as well as the assumption that $\mathcal{G}$ is a pseudorandom generator and $\mathcal{F}$ is XOR-IND (as defined in Section 7.1).

Theorem 7.4 Let η, μ, ζ, ρ be integer parameters. Let $\mathcal{F}$ be a family of trapdoor permutations with domain $\{0,1\}^\mu$, and let $\eta = |[\mathsf{Kg}(1^k)]|$. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be function families. Suppose $\mathcal{G}$ is a pseudorandom generator, and let $\mathcal{F}$ is XOR-IND0 with respect to hint function $\mathcal{G}$ (as defined in Section 7.1). Also suppose $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}]$ is η-EXT1$^{\mu+\zeta}$ and NCR$^{\mu+\zeta}$. Then $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|^{\mu+\zeta}]$ is IND-CCA1 secure. In particular, for any adversary A that makes q decryption queries, there exist adversaries C, D, E, and EXT1 adversary B that makes q extract queries such that for all extractors Ext,

$$\begin{aligned}\mathbf{Adv}^{\text{ind-cca1}}_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}},A}(k) \quad &\leq \quad 2 \cdot \mathbf{Adv}^{\eta\text{-ext1}^{\mu+\zeta}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H}],B,\mathsf{Ext}}(k) + 2 \cdot \mathbf{Adv}^{\text{n-cr}^{\mu+\zeta}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H}],C}(k) \\ &\quad +6 \cdot \mathbf{Adv}^{\text{xor-ind0}}_{\mathcal{F},\mathcal{G},D}(k) + 4 \cdot \mathbf{Adv}^{\text{prg}}_{\mathcal{G},E}(k) \ .\end{aligned}$$

IND-CCA2 RESULT. To prove IND-CCA2, we use EXT2 and near-collision resistance of $\mathcal{G}$, as well as the assumptions that $\mathcal{G}$ is a pseudorandom generator, $\mathcal{H}$ is collision-resistant and $\mathcal{F}$ is XOR-IND and XOR-NM (as defined in Section 7.1). Note that, EXT2 adversary only makes one image query. Thus, the dependent auxiliary information is bounded by the size of the image.

Theorem 7.5 Let η, μ, ζ, ρ be integer parameters. Let $\mathcal{F}$ be a family of trapdoor permutations with domain $\{0,1\}^\mu$ and $\eta = |[\mathsf{Kg}(1^k)]| + |[\mathcal{K}_H(1^k)]|$. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be function families. Suppose $\mathcal{G}$ is PRG, NCR$_\zeta$, EXT2$_\zeta$ and η-EXT2$_\zeta$ with respect to $\mathcal{F}$, and $\mathcal{H}$ is collision-resistant. Suppose $\mathcal{F}$ is XOR-NM0, XOR-IND1 and XOR-IND2$_\zeta$ with respect to $\mathcal{G}$. Then $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|^{\mu+\zeta}]$ is IND-CCA2 secure. In particular, for any adversary A that makes q decryption queries, there exists adversaries $C_H, C_G, D_1, D_2, D_3, E$, and adversary B_1, B_2 that makes q extract queries such that for all extractors $\mathsf{Ext}_1, \mathsf{Ext}_2$,

$$\begin{aligned}\mathbf{Adv}^{\text{ind-cca2}}_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}},A}(k) \quad &\leq \quad 6 \cdot \mathbf{Adv}^{\eta\text{-ext2}_\zeta}_{\mathcal{G},\mathcal{F},B_1,\mathsf{Ext}_1}(k) + 18 \cdot \mathbf{Adv}^{\text{xor-ind2}_\zeta}_{\mathcal{F},\mathcal{G},D_1}(k) \\ &\quad +10 \cdot \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G},C_G}(k) + 4 \cdot \mathbf{Adv}^{\text{cr}}_{\mathcal{H},C_H}(k) + 4 \cdot \mathbf{Adv}^{\text{xor-nm0}}_{\mathcal{F},\mathcal{G},D_3}(k) \\ &\quad +14 \cdot \mathbf{Adv}^{\text{xor-ind1}}_{\mathcal{F},\mathcal{G},D_2}(k) + 16 \cdot \mathbf{Adv}^{\text{prg}}_{\mathcal{G},E}(k) + 24 \cdot \mathbf{Adv}^{\text{ext2}_\zeta}_{\mathcal{G},B_2,\mathsf{Ext}_2}(k) \ .\end{aligned}$$

EFFICIENCY. The ciphertext length is $2n + k + \mu$ where n is the length of the RSA modulus, k is the security parameter, and μ is the message length. For example, if $n = 2048$, $k = 128$, and we encrypt an AES key with $\mu = 128$ (*i.e.*, we use RSA-OAEP as a key encapsulation mechanism, which is typical in practice then the ciphertext length is 4352. It is interesting to compare this with the standard model IND-CCA2 secure key encapsulation mechanism of Kiltz *et al.* [53]. They describe their scheme based on modular squaring (factoring), but it is straightforward to derive a scheme based on RSA with large hardcore function and a cryptographic hash function being target collision-resistant, which results in the most efficient prior standard-model RSA-based encryption scheme we are aware of. It performs one "small" exponentiation wrt. e and one "full" exponentiation modulo N, so is much more computationally expensive than our scheme. Thus, one could arguably say ours is the most computationally efficient RSA-based encryption scheme under "plausible standard-model assumptions" (where one takes the liberty of making bold assumptions on cryptographic hash functions) to date. On the other hand, the scheme of [53] has ciphertext length only $2n$.

Games $G_1(k)$, $G_2(k)$	**Games** $G_3(k)$
$b \leftarrow\!\!\$\ \{0,1\}$	$b \leftarrow\!\!\$\ \{0,1\}$
$K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$; $K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$	$K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$; $K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	$f \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$
$pk \leftarrow (K_G, K_H, f)$	$pk \leftarrow (K_G, K_H, f)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1(1^k, pk)$	$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1(1^k, pk)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b(1^k, pk)$; $r \leftarrow\!\!\$\ \{0,1\}^\rho$	$m_b \leftarrow\!\!\$\ \mathcal{M}_b(1^k, pk)$; $r \leftarrow\!\!\$\ \{0,1\}^\rho$
$x \leftarrow G(K_G, r)$; $s \leftarrow x \oplus (m_b \| 0^\zeta)$	$x \leftarrow \{0,1\}^{\mu+\zeta}$; $s \leftarrow x \oplus (m_b \| 0^\zeta)$
$z \leftarrow H(K_H, s)$; $t \leftarrow z \oplus r$	$y \leftarrow f(r)$; $c \leftarrow (s, y)$
$y \leftarrow f(t)$; $y \leftarrow f(r)$	$b' \leftarrow\!\!\$\ A_2(state, c)$
$c \leftarrow (s, y)$; $b' \leftarrow\!\!\$\ A_2(state, c)$	Return $(b = b')$
Return $(b = b')$	

Figure 50: **Games G_1–G_3 in the proof of Theorem 7.7.**

Remark 7.6 It is worth mentioning why we are able to get IND-CCA2 (*i.e.*, adaptive) security for s-clear RSA-OAEP but not t-clear. The point is that, in the t-clear setting, it is not even clear how to define EXT2 of OAEP in a useful way. Since OAEP is invertible, the image oracle should output only *part* of the image point. But then it is not clear how the EXT2 adversary against OAEP can simulate the encryption oracle for the PA2 adversary against t-clear RSA-OAEP. On the other hand, for EXT2 of $\mathcal{G}$, the image oracle can output the *full* image point since $\mathcal{G}$ is not invertible. This then allows proving that s-clear RSA-OAEP is PA2 directly (without using monolithic assumptions on the padding scheme not known to follow from assumptions on the round functions).

7.3 IND-CPA Result

We first show that s-clear RSA-OAEP is IND-CPA secure under suitable assumptions. Then, we show PA0, PA1 and PA2 security depend on the strength of assumptions on $\mathcal{G}, \mathcal{H}$ and $\mathcal{F}$. Interestingly, our IND-CPA result uses an XOR-based assumption on the trapdoor permutation.

Theorem 7.7 Let μ, ζ, ρ be integer parameters. Let $\mathcal{F}$ be a family of trapdoor permutations with domain $\{0,1\}^\rho$. Suppose $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ is a pseudorandom generator and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ is a family of hash function. Suppose $\mathcal{F}$ is XOR-IND0 with respect to hint function family $\mathcal{G}$. Then $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|^{\mu+\zeta}]$ is IND-CPA secure. In particular, for any adversary A, there are adversaries B, D such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}},A}(k) \le 6 \cdot \mathbf{Adv}^{\text{xor-ind0}}_{\mathcal{F},\mathcal{G},B}(k) + 4 \cdot \mathbf{Adv}^{\text{prg}}_{\mathcal{G},D}(k) \ .$$

The running time of B and D are about that of A.

Proof: Consider games G_1–G_3 in Figure 50. Game G_1 corresponds to game $\text{IND-CPA}^A_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}}}$. We now explain the game chain. Game G_2 is identical to game G_1, except that instead of evaluating trapdoor permutation f on input t, we are evaluating it on input r. Consider adversary B as shown in Figure 51. Note that adversary B simulates game G_1, G_2 with respect to it's inputs. It returns 1 if adversary A can correctly guess the simulated challenge bit b, and returns 0 otherwise. Hence,

$$\Pr[G_1 \Rightarrow 1] - \Pr[G_2 \Rightarrow 1] \le \mathbf{Adv}^{\text{xor-ind0}}_{\mathcal{F},\mathcal{G},B}(k) \ .$$

Next, game G_3 is identical to game G_2, except we are using completely random x in the encryption phase instead of using pseudorandom value $G(K_G, r)$. Consider adversary C as shown in Figure 52. Hence,

$$\Pr[G_2 \Rightarrow 1] - \Pr[G_3 \Rightarrow 1] \le \mathbf{Adv}^{\text{hcf}}_{\mathcal{F},\mathcal{G},U_\rho,C}(k) \ .$$

From Theorem 7.1, there are adversaries D, E such that

$$\Pr[G_2 \Rightarrow 1] - \Pr[G_3 \Rightarrow 1] \le 2 \cdot \mathbf{Adv}^{\text{xor-ind0}}_{\mathcal{F},\mathcal{G},E}(k) + 2 \cdot \mathbf{Adv}^{\text{prg}}_{\mathcal{G},D}(k) \ .$$

We assume wlog that advantage of adversary B is greater than adversary E. Note that $\Pr[G_3(k) \Rightarrow 1] = 1/2$,

Algorithm $B_1(f, K_G, G(K_G, r))$	**Algorithm** $B_2(state, y_b)$
$b \leftarrow\!\!\$\ \{0,1\}$; $K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$	$(b, pk, s) \leftarrow state$
$pk \leftarrow (K_G, K_H, f)$	$c \leftarrow (s, y_b)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1(1^k, pk)$	$b' \leftarrow\!\!\$\ A_2(state, c)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b(1^k, pk)$	Return $(b = b')$
$s \leftarrow G(K_G, r) \oplus (m_b \| 0^\zeta)$	
$z \leftarrow H(K_H, s)$; $state \leftarrow (b, pk, s)$	
Return $(state, z)$	

Figure 51: **Adversary B in the proof of Theorem 7.7.**

Algorithm $C(K_G, f, f(r), x)$
$b \leftarrow\!\!\$\ \{0,1\}$; $K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$
$pk \leftarrow (K_G, K_H, f)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1(1^k, pk)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b(1^k, pk)$
$s \leftarrow x \oplus (m_b \| 0^\zeta)$; $c \leftarrow (s, f(r))$
$b' \leftarrow\!\!\$\ A_2(state, c)$
Return $(b = b')$

Figure 52: **Adversary C in the proof of Theorem 7.7.**

since the distribution of the ciphertexts is completely independent of bit b. Summing up,

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}_{\mathsf{s-clear}},A}(k) \le 6 \cdot \mathbf{Adv}^{\text{xor-ind0}}_{\mathcal{F},\mathcal{G},B}(k) + 4 \cdot \mathbf{Adv}^{\text{prg}}_{\mathcal{G},D}(k) \ .$$

This completes the proof. ∎

7.4 PA0 and PA1 Results

We give a full instantiation result for s-clear RSA-OAEP and show that it is PA0 and PA1 under suitable assumptions. We show that s-clear RSA-OAEP "inherits" the extractability of the underlying padding transform, in the form of PA0 and PA1, as long as the latter is also near-collision resistant. Here we state the result for an abstract padding scheme rather than specifically for OAEP. Note that results for OAEP then follow from the round functions in Section 5.

Theorem 7.8 Let η, μ, ζ, ρ be integer parameters. Let $\mathcal{F}$ be a family of one-way trapdoor permutations with domain $\{0,1\}^\rho$ and $\eta = |[\mathsf{Kg}(1^k)]|$. Let PAD be a padding transform from domain $\{0,1\}^{\mu+\rho}$ to range $\{0,1\}^{\mu+\zeta+\rho}$. Suppose PAD is $\text{NCR}^{\mu+\zeta}$ and η-$\text{EXT0}^{\mu+\zeta}$. Then $\mathsf{PAD}[\mathcal{F}|^{\mu+\zeta}]$ is PA0 secure. In particular, for any PA0 adversary A, there are adversaries A_{PAD}, B and extractor Ext such that for all extractors $\mathsf{Ext}_{\mathsf{PAD}}$

$$\mathbf{Adv}^{\text{pa0}}_{\mathsf{PAD}[\mathcal{F}|^{\mu+\zeta}],A,\mathsf{Ext}}(k) \le \mathbf{Adv}^{\eta\text{-ext0}^{\mu+\zeta}}_{\mathsf{PAD},A_{\mathsf{PAD}},\mathsf{Ext}_{\mathsf{PAD}}}(k) + \mathbf{Adv}^{\text{n-cr}^{\mu+\zeta}}_{\mathsf{PAD},B}(k) \ .$$

The running time of A_{PAD} and Ext are about that of A and $\mathsf{Ext}_{\mathsf{PAD}}$, respectively. Furthermore, the running time of B is about that of A plus the time to run $\mathsf{Ext}_{\mathsf{PAD}}$.

The proof of Theorem 7.8 is very similar to the proof of Theorem 7.9.

Theorem 7.9 Let η, μ, ζ, ρ be integer parameters. Let $\mathcal{F}$ be a family of one-way trapdoor permutations with domain $\{0,1\}^\rho$ and $\eta = |[\mathsf{Kg}(1^k)]|$. Let PAD be a padding transform from domain $\{0,1\}^{\mu+\rho}$ to range $\{0,1\}^{\mu+\zeta+\rho}$. Suppose PAD is $\text{NCR}^{\mu+\zeta}$ and η-$\text{EXT1}^{\mu+\zeta}$. Then $\mathsf{PAD}[\mathcal{F}|^{\mu+\zeta}]$ is PA1 secure. In particular, for any PA1 adversary A that makes at most q decryption queries, there are adversaries B, A_{PAD} that makes at most q extract queries and extractor Ext such that for all extractors $\mathsf{Ext}_{\mathsf{PAD}}$

$$\mathbf{Adv}^{\text{pa1}}_{\mathsf{PAD}[\mathcal{F}|^{\mu+\zeta}],A,\mathsf{Ext}}(k) \le \mathbf{Adv}^{\eta\text{-ext1}^{\mu+\zeta}}_{\mathsf{PAD},A_{\mathsf{PAD}},\mathsf{Ext}_{\mathsf{PAD}}}(k) + \mathbf{Adv}^{\text{n-cr}^{\mu+\zeta}}_{\mathsf{PAD},B}(k) \ .$$

The running time of A_{PAD} and Ext are about that of A and $\mathsf{Ext}_{\mathsf{PAD}}$, respectively. Furthermore, the running time of B is about that of A plus the time to run $\mathsf{Ext}_{\mathsf{PAD}}$.

Adversary $A_{\mathsf{PAD}}^{\mathcal{O}_{\mathsf{PAD}}(\cdot)}(\pi, f; w)$
$pk \leftarrow (\pi, f)$
Run $A^{\mathcal{D}\textsc{Sim}(\cdot)}(pk; w)$

Procedure $\mathcal{D}\textsc{Sim}(c)$
$v \leftarrow \mathcal{O}_{\mathsf{PAD}}(\bot, c|^{\mu+\zeta})$
Return $v|^{\mu}$

Figure 53: **EXT1 adversary A_{PAD} in the proof of Theorem 7.9.**

Algorithm $\mathsf{Ext}(state, c)$
$(pk, w, st) \leftarrow state$
$(v, st) \leftarrow \mathsf{Ext}_{\mathsf{PAD}}(st, \pi, f, c|^{\mu+\zeta}; w)$
$state \leftarrow (pk, w, st)$
$m \leftarrow v|^{\mu}$; $r \leftarrow v|_{\rho}$
If $c \neq \mathsf{Enc}(pk, m; r)$ then return $(state, \bot)$
Return $(state, m)$

Figure 54: **PA1 extractor Ext in the proof of Theorem 7.9.**

Proof: Let w be the randomness of PA1 adversary A. We define EXT1 adversary A_{PAD} with randomness w in Figure 53. Note that the auxiliary input f is independent of the key π. Let $\mathsf{Ext}_{\mathsf{PAD}}$ be the corresponding extractor for A_{PAD}. We define PA1 extractor Ext as shown in Figure 54.

Note that for any decryption query c that adversary A makes, if c is not a valid ciphertext then extractor Ext outputs $\bot$. Thus, adversary A does not gain any information about b by making invalid decryption queries. Hence, we assume wlog that adversary A only makes queries for valid ciphertext c.

Assume A makes q extract queries. Let c_i be the i-th query A makes to the extract oracle and v_i be the output of extractor $\mathsf{Ext}_{\mathsf{PAD}}$ on input $c_i|^{\mu+\zeta}$. Let W be the event where there exists a valid ciphertext c_i such that v_i is non-empty and extractor Ext outputs $\bot$ on input c_i. Let S be the event that game $\mathrm{PA1}^{A,\mathsf{Ext}}_{\mathsf{PAD}[\mathcal{F}|^{\mu+\zeta}]}(k)$ outputs 1. Note that all of the following probabilities are over the choice of public key pk and randomness w. Then,

$$\mathbf{Adv}^{\mathrm{pa1}}_{\mathsf{PAD}[\mathcal{F}|^{\mu+\zeta}],A,\mathsf{Ext}}(k) = 2 \cdot \big(\Pr[\, S \wedge W \,] + \Pr\big[\, S \wedge \overline{W} \,\big]\big) - 1 \ .$$

Note that extractor $\mathsf{Ext}_{\mathsf{PAD}}$ either outputs the correct value or $\bot$. Now consider near-collision resistance adversary B in Figure 55. If event W happens, then B finds a collision. Thus, we have $\Pr[\, W \,] = \mathbf{Adv}^{\text{n-cr}^{\mu+\zeta}}_{\mathsf{PAD},B}(k)$. Then

$$\Pr[\, S \wedge W \,] \leq \mathbf{Adv}^{\text{n-cr}^{\mu+\zeta}}_{\mathsf{PAD},B}(k) \ .$$

Let E be the event such that for each decryption query c_i that adversary A makes, the output of decryption algorithm Dec and extractor Ext are equal. Then, we obtain $\Pr\big[\, S \wedge \overline{W} \,\big] = \Pr\big[\, S \wedge \overline{W} \wedge E \,\big] + \Pr\big[\, S \wedge \overline{W} \wedge \overline{E} \,\big]$. Note that W and E are mutually exclusive since when W happens the output of extractor Ext would be incorrect for at least one of the extract queries. Thus, we have $\Pr\big[\, S \wedge \overline{W} \wedge E \,\big] = \Pr[\, S \wedge E \,]$. Moreover, we have $\Pr[\, S \mid E \,] = 1/2$, since for all queries made by A, the outputs of decryption oracle Dec and extractor Ext are equal. Hence, we obtain $\Pr[\, S \wedge E \,] = 1/2 \cdot \Pr[\, E \,]$.

Consider EXT1 adversary A_{PAD} in Figure 53. We know A always makes valid decryption query. Thus, when event $\overline{E}$ happens Ext outputs $\bot$ for at least one of the extract queries. Moreover, if $\overline{W}$ happens then for all extract queries either $\mathsf{Ext}_{\mathsf{PAD}}$ $\bot$ or Ext outputs non-empty string. Therefore, when $\overline{E}$ and $\overline{W}$ happen, extractor $\mathsf{Ext}_{\mathsf{PAD}}$ fails and outputs $\bot$ for at least one of the extract queries. Thus,

$$\Pr\big[\, \overline{W} \wedge \overline{E} \,\big] \leq \mathbf{Adv}^{\eta\text{-ext1}^{\mu+\zeta}}_{\mathsf{PAD},A_{\mathsf{PAD}},\mathsf{Ext}_{\mathsf{PAD}}}(k) \ .$$

On the other hand, we know that E and W mutually exclusive. Hence, we get $\Pr[\, E \,] = \Pr[\, W \vee E \,] - \Pr[\, W \,]$. Summing up,

$$\mathbf{Adv}^{\mathrm{pa1}}_{\mathsf{PAD}[\mathcal{F}|^{\mu+\zeta}],A,\mathsf{Ext}}(k) \leq \mathbf{Adv}^{\eta\text{-ext1}^{\mu+\zeta}}_{\mathsf{PAD},A_{\mathsf{PAD}},\mathsf{Ext}_{\mathsf{PAD}}}(k) + \mathbf{Adv}^{\text{n-cr}^{\mu+\zeta}}_{\mathsf{PAD},B}(k) \ .$$

Adversary $B(\pi, \hat{\pi})$	**Procedure** $\mathcal{D}\text{SIM}(c)$
$\mathsf{out}_1 \leftarrow \varepsilon$; $\mathsf{out}_2 \leftarrow \varepsilon$	$y_1 \leftarrow c\|^{\mu+\zeta}$; $y_2 \leftarrow f^{-1}(c\|_\rho)$
$w \leftarrow\!\!\$\ \mathsf{Coins}(k)$; $b \leftarrow\!\!\$\ \{0,1\}$	$y \leftarrow y_1 \| y_2$; $\mathsf{out}_1 \leftarrow \hat{\pi}(y)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	$(st, \mathsf{out}_2) \leftarrow \mathsf{Ext}_{\mathsf{PAD}}(st, \pi, f, \perp, c\|^{\mu+\zeta}; w)$
$pk \leftarrow (\pi, f)$; $sk \leftarrow (\hat{\pi}, f^{-1})$	If $\big(\pi(\mathsf{out}_1)\|^{\mu+\zeta} = \pi(\mathsf{out}_2)\|^{\mu+\zeta} \wedge \mathsf{out}_1 \neq \mathsf{out}_2\big)$
$st \leftarrow (\pi, w)$; $state \leftarrow (pk, w)$	Halt A
Run $A^{\mathcal{D}\text{SIM}(\cdot)}(pk; w)$	$m_0 \leftarrow \mathsf{Dec}(sk, c)$
Return $(\mathsf{out}_1, \mathsf{out}_2)$	$(state, m_1) \leftarrow\!\!\$\ \mathsf{Ext}(state, c)$
	Return m_b

Figure 55: **NCR adversary B in the proof of Theorem 7.9.**

This completes the proof. ∎

7.5 PA2 Result

We give a full instantiation result for s-clear RSA-OAEP and show that it is PA2 under stronger assumptions on $\mathcal{G}, \mathcal{H}$ and $\mathcal{F}$. We note that we can reduce assumptions as per Theorem 7.2.

Theorem 7.10 Let η, μ, ζ, ρ be integer parameters. Let $\mathcal{F}$ be a family of trapdoor permutations with domain $\{0,1\}^\rho$. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be hash function families. Let $\eta = |[\mathsf{Kg}(1^k)]| + |[\mathcal{K}_H(1^k)]|$. Suppose $\mathcal{G}$ is PRG, NCR_ζ, EXT2_ζ and η-EXT2_ζ with respect to $\mathcal{F}$ and $\mathcal{H}$ is collision-resistant. Suppose $\mathcal{F}$ is XOR-NM1 and XOR-IND2_ζ with respect to $\mathcal{G}$. Then $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|^{\mu+\zeta}]$ is PA2 secure. In particular, for any adversary A that makes at most q decryption queries and p encryption queries, there are extractor Ext, adversaries B_F, B_G, B_H, C, D, adversary A_G, C_G that makes at most q extract queries and p image queries such that for all extractor $\mathsf{Ext}_G, \mathsf{Ext}'_G$

$$\begin{aligned}\mathbf{Adv}^{\text{pa2}}_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}}, A, \mathsf{Ext}}(k) \;\leq\; & 3 \cdot \mathbf{Adv}^{\eta\text{-ext2}_\zeta}_{\mathcal{G}, \mathcal{F}, A_G, \mathsf{Ext}_G}(k) + 9p \cdot \mathbf{Adv}^{\text{xor-ind2}_\zeta}_{\mathcal{F}, \mathcal{G}, C}(k) + 6p \cdot \mathbf{Adv}^{\text{prg}}_{\mathcal{G}, D}(k) \\ & + 12p \cdot \mathbf{Adv}^{\text{ext2}_\zeta}_{\mathcal{G}, C_G, A_G, \mathsf{Ext}'_G}(k) + 5 \cdot \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G}, B_G}(k) + 2 \cdot \mathbf{Adv}^{\text{cr}}_{\mathcal{H}, B_H}(k) + 2p \cdot \mathbf{Adv}^{\text{xor-nm1}}_{\mathcal{F}, \mathcal{G}, B_F}(k)\end{aligned}$$

The running time of A_G, C_G is about that of A. The running time of Ext is about that of Ext_G. The running time of C and D are about that of A plus the time to run Ext_G. The running time of B_F, B_G and B_H are about that of A plus the time to run Ext'_G.

Proof: We will need Lemma 7.11 in our proof. We refer to Appendix E for the proof of Lemma 7.11.

Lemma 7.11 Let $\eta, \delta, \mu, \zeta, \rho$ be integer parameters. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be a hash function family. Let $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a trapdoor permutation family with domain $\{0,1\}^\rho$ and $\eta = \delta + |[\mathcal{K}_H(1^k)]|$. Suppose $\mathcal{G}$ is VPRG_ζ and η-EXT2_ζ function with respect to $\mathcal{F}$. Suppose $\mathcal{F}$ is XOR-IND2_ζ with respect to $\mathcal{G}$. Then $\mathcal{G}$ is a δ-EXT2_ζ with respect to $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|^{\mu+\zeta}]$. In particular, for any adversary A that makes at most q extract queries and p image queries, there are adversaries C, D and adversary B that makes at most q extract queries and p image queries such that for all extractor Ext

$$\mathbf{Adv}^{\delta\text{-ext2}_\zeta}_{\mathcal{G}, \mathsf{OAEP}_{\mathsf{s\text{-}clear}}, A, \mathsf{Ext}}(k) \leq \mathbf{Adv}^{\eta\text{-ext2}_\zeta}_{\mathcal{G}, \mathcal{F}, B, \mathsf{Ext}}(k) + 3p \cdot \mathbf{Adv}^{\text{xor-ind2}_\zeta}_{\mathcal{F}, \mathcal{G}, C}(k) + 2p \cdot \mathbf{Adv}^{\text{vprg}_\zeta}_{\mathcal{G}, D}(k) \ .$$

The running time of B is about that of A. The running time of C and D are about that of A plus the time to run Ext.

Note that we didn't define EXT2 function with respect to encryption scheme $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|^{\mu+\zeta}]$. We point that the definition of such a notion is very similar to the original EXT2 notion. Moreover, we only use this notion inside this proof and we show in Lemma 7.11 that it is implied by original EXT2 notion under certain assumption. We now show in Lemma 7.12 that when a function is PRG and EXT2, then it is also VPRG. We refer to Appendix E for the proof of Lemma 7.12.

Lemma 7.12 Let μ, ρ, δ, ζ be integer parameters. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ be a hash function family. Suppose $\mathcal{G}$ is PRG and δ-EXT2_ζ function, for any $\delta > 0$. Then $\mathcal{G}$ is a VPRG_ζ. In particular, for any adversary A

Adversary $A_G^{\mathcal{O}_G(\cdot),\mathcal{I}(\cdot)}(K_G, v; w)$ $(K_H, f) \leftarrow v$; $pk \leftarrow (K_G, K_H, f)$ Run $A^{\mathcal{D}\text{Sim}(\cdot),\text{EncSim}(\cdot)}(pk; w)$ **Procedure** EncSim($\mathcal{M}$) $(x, c) \leftarrow \mathcal{I}(\mathcal{M})$ Return c	**Procedure** $\mathcal{D}$Sim(c) $(s, y) \leftarrow c$; $r \leftarrow \mathcal{O}_G(s\|_\zeta)$ $x \leftarrow G(K_G, r)$ $m^* \leftarrow s \oplus x$; $m \leftarrow m^*\|^\mu$ Return m

Figure 56: **EXT2 adversary A_G in the proof of Theorem 7.10.**

Algorithm $\mathsf{Ext}(state, \mathbf{c}, c)$
$(pk, w, st) \leftarrow state$
For $i = 1$ to $|\mathbf{c}|$ do
  $(\mathbf{s}[i], \mathbf{y}[i]) \leftarrow \mathbf{c}[i]$; $\mathbf{s}_2[i] \leftarrow \mathbf{s}[i]|_\zeta$
$(s, y) \leftarrow c$; $(K_G, K_H, f) \leftarrow pk$
$v \leftarrow (K_H, f)$
$(st, r) \leftarrow \mathsf{Ext}_G(st, K_G, v, \mathbf{s}_2, \mathbf{c}, s|_\zeta; w)$
$state \leftarrow (pk, w, st)$; $x \leftarrow G(K_G, r)$
$m^* \leftarrow s \oplus x$; $m \leftarrow m^*\|^\mu$
If $m^*|_\zeta \neq 0^\zeta$ then return $(\perp, state)$
If $c \neq \mathsf{Enc}(pk, m; r)$ then return $(\perp, state)$
Return $(m, state)$

Figure 57: **PA2 extractor Ext in the proof of Theorem 7.10.**

that makes at most q verification queries, there are adversary D and adversary B that makes at most q extract queries and one image query such that for all extractor Ext

$$\mathbf{Adv}^{\text{vprg}_\zeta}_{\mathcal{G},A}(k) \leq \mathbf{Adv}^{\text{prg}}_{\mathcal{G},D}(k) + 2 \cdot \mathbf{Adv}^{\delta\text{-ext2}_\zeta}_{\mathcal{G},B,\mathsf{Ext}}(k) \ .$$

Let w be the randomness of adversary A in the PA2 game. Let K_H be the key for the hash function family H and f be the evaluation key for the trapdoor permutation family $\mathcal{F}$ in the game PA2. We define EXT2 adversary A_G with the hint function $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}$ and randomness w in Figure 56. Let $v = (K_H, f)$ be the key independent auxiliary input to adversary A_G. Note that auxiliary input v is independent of key K_G. Let Ext_G be the corresponding extractor for A_G. We define PA2 extractor Ext as shown in Figure 57. Let $c_i = (s_i, y_i)$ be the i-th decryption query made by A and r_i be the correspondence randomness. Note that when decryption query c_i is invalid Ext will always output $\perp$. Hence, the output of Ext and decryption algorithm are always equal on invalid decryption queries. Thus, we only consider valid queries c_i made by A.

We define C_i to be the set of all ciphertext c produced by encryption oracle before adversary A makes the i-th decryption query, for all $i \in [q]$. We also define $C_{i,1}$ to be the set of all $c \in C_i$ such that $s_i|_\zeta = s|_\zeta$. Let S be the event that game $\text{PA2}^{A,\mathsf{Ext}}_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}}}(k)$ outputs 1 and E be the event such that for all decryption query c_i made by A, $C_{i,1}$ is empty. Then

$$\mathbf{Adv}^{\text{pa2}}_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}},A,\mathsf{Ext}}(k) = 2 \cdot \big(\Pr[\, S \wedge E \,] + \Pr[\, S \wedge \overline{E} \,]\big) - 1 \ .$$

Note that using the same argument made in the proof of Theorem 7.9, there exists adversaries A_G, B such that for $\delta = |[\mathsf{Kg}(1^k)]|$, we have

$$\Pr[\, S \wedge E \,] \leq \frac{1}{2} \cdot \big(\mathbf{Adv}^{\delta\text{-ext2}_\zeta}_{\mathcal{G},\mathsf{OAEP}_{\mathsf{s\text{-}clear}},A_G,\mathsf{Ext}_G}(k) + \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G},B}(k) + 1\big) \ .$$

Let W be the event such that for at least one decryption query c_i, there exists $c \in C_{i,1}$ where $r_i \neq r$. Then, we obtain $\Pr[\, \overline{E} \,] \leq \Pr[\, \overline{E} \wedge W \,] + \Pr[\, \overline{E} \wedge \overline{W} \,]$. Consider near collision resistance adversary B_G in Figure 58. Note that, when W and $\overline{E}$ happen, adversary B_G finds a collision. Thus,

$$\Pr[\, \overline{E} \wedge W \,] \leq \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G},B_G}(k) \ .$$

Adversary $B_G(K_G)$	**Procedure** $\mathcal{D}\text{SIM}(c)$	**Procedure** $\text{ENCSIM}(\mathcal{M})$
$b \leftarrow \{0,1\}$; $i \leftarrow 1$	$(s,y) \leftarrow c$; $t \leftarrow f^{-1}(y)$	$m \leftarrow \mathcal{M}(1^k, pk)$
$\mathbf{r} \leftarrow \varepsilon$; $\mathbf{c} \leftarrow \varepsilon$; $\mathbf{s} \leftarrow \varepsilon$	$\mathsf{out}_1 \leftarrow t \oplus H(K_H, s)$	$\mathbf{r}[i] \leftarrow\!\!\$\ \{0,1\}^\rho$
$\mathsf{out}_1 \leftarrow \varepsilon$; $\mathsf{out}_2 \leftarrow \varepsilon$	For $i = 1$ to $\lvert\mathbf{c}\rvert$ do	$\mathbf{c}[i] \leftarrow \mathsf{Enc}(pk, m; \mathbf{r}[i])$
$st \leftarrow \varepsilon$; $w \leftarrow\!\!\$\ \mathsf{Coins}(k)$	If $s\lvert_\zeta = \mathbf{s}[i]\rvert_\zeta \wedge \mathsf{out}_1 \neq \mathbf{r}[i]$	$\mathbf{s}[i] \leftarrow \mathbf{c}[i]\rvert^{\mu+\zeta}$; $i \leftarrow i+1$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	$\mathsf{out}_2 \leftarrow \mathbf{r}[i]$; Halt A	Return $\mathbf{c}[i]$
$K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$	$m_0 \leftarrow \mathsf{Dec}(sk, c)$	
$pk \leftarrow (K_G, K_H, f)$	$(m_1, state) \leftarrow\!\!\$\ \mathsf{Ext}(state, \mathbf{c}, c)$	
$sk \leftarrow (K_G, K_H, f^{-1})$	Return m_b	
$state \leftarrow (pk, w)$		
Run $A^{\mathcal{D}\text{SIM}(\cdot), \text{ENCSIM}(\cdot)}(pk; w)$		
Return $(\mathsf{out}_1, \mathsf{out}_2)$		

Figure 58: **NCR adversary B_G in the proof of Theorem 7.10.**

Adversary $B_H(K_H)$	**Procedure** $\mathcal{D}\text{SIM}(c)$	**Procedure** $\text{ENCSIM}(\mathcal{M})$
$b \leftarrow \{0,1\}$; $i \leftarrow 1$	$(s,y) \leftarrow c$; $\mathsf{out}_1 \leftarrow s$	$m \leftarrow \mathcal{M}(1^k, pk)$
$\mathbf{r} \leftarrow \varepsilon$; $\mathbf{c} \leftarrow \varepsilon$; $\mathbf{y} \leftarrow \varepsilon$	$t \leftarrow f^{-1}(y)$; $r \leftarrow t \oplus H(K_H, s)$	$\mathbf{r}[i] \leftarrow\!\!\$\ \{0,1\}^\rho$
$\mathsf{out}_1 \leftarrow \varepsilon$; $\mathsf{out}_2 \leftarrow \varepsilon$	For $i = 1$ to $\lvert\mathbf{c}\rvert$ do	$\mathbf{c}[i] \leftarrow \mathsf{Enc}(pk, m; \mathbf{r}[i])$
$st \leftarrow \varepsilon$; $w \leftarrow\!\!\$\ \mathsf{Coins}(k)$	If $r = \mathbf{r}[i] \wedge y = \mathbf{y}[i]$	$\mathbf{y}[i] \leftarrow \mathbf{c}[i]\rvert_\rho$; $i \leftarrow i+1$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$	If $\mathsf{out}_1 \neq \mathbf{c}[i]\rvert^{\mu+\zeta}$	Return $\mathbf{c}[i]$
$K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$	$\mathsf{out}_2 \leftarrow \mathbf{c}[i]\rvert^{\mu+\zeta}$; Halt A	
$pk \leftarrow (K_G, K_H, f)$	$m_0 \leftarrow \mathsf{Dec}(sk, c)$	
$sk \leftarrow (K_G, K_H, f^{-1})$	$(m_1, state) \leftarrow\!\!\$\ \mathsf{Ext}(state, \mathbf{c}, c)$	
$state \leftarrow (pk, w)$	Return m_b	
Run $A^{\mathcal{D}\text{SIM}(\cdot), \text{ENCSIM}(\cdot)}(pk; w)$		
Return $(\mathsf{out}_1, \mathsf{out}_2)$		

Figure 59: **CR adversary B_H in the proof of Theorem 7.10.**

Moreover, let Q be the event such that for at least one decryption query c_i, there exists $c \in C_{i,1}$ where $y_i = y$. Then, we get that $\Pr\left[\,\overline{E} \wedge \overline{W}\,\right] \leq \Pr\left[\,\overline{E} \wedge \overline{W} \wedge Q\,\right] + \Pr\left[\,\overline{E} \wedge \overline{W} \wedge \overline{Q}\,\right]$. Consider collision resistance adversary B_H in Figure 59. Note that, when events $\overline{E}, \overline{W}$ and Q happen, adversary B_H finds a collision. Thus,

$$\Pr\left[\,\overline{E} \wedge \overline{W} \wedge Q\,\right] \leq \mathbf{Adv}^{\text{cr}}_{\mathcal{H}, B_H}(k) \ .$$

Note that when events $\overline{E}, \overline{W}$ and $\overline{Q}$ happen, for all decryption query c_i with non-empty $C_{i,1}$, we have that $r_i = r$ and $y_i \neq y$ for all $c = (s,y) \in C_{i,1}$. Let c_j be the first decryption query with non-empty $C_{j,1}$. Let R be the event such that for all decryption query c_i with $i < j$ the output of Ext and decryption algorithm are equal. Therefore, using the same argument made in the proof of Theorem 7.9, there exists adversary B such that

$$\Pr\left[\,\overline{E} \wedge \overline{W} \wedge \overline{Q} \wedge \overline{R}\,\right] \leq \mathbf{Adv}^{\delta\text{-ext2}_\zeta}_{\mathcal{G}, \mathsf{OAEP}_{\mathsf{s-clear}}, A_G, \mathsf{Ext}_G}(k) + \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G}, B}(k) \ .$$

Consider XOR-NM1 adversary B_F in Figure 60. Note that, when events $\overline{E}, \overline{W}$ $\overline{Q}$ and R happen, adversary B_F can win the XOR-NM1 game. Thus, we obtain $\Pr\left[\,\overline{E} \wedge \overline{W} \wedge \overline{Q} \wedge R\,\right] \leq p \cdot \mathbf{Adv}^{\text{xor-nm1}}_{\mathcal{F}, \mathcal{G}, B_F}(k)$. Summing up,

$$\begin{aligned} \mathbf{Adv}^{\text{pa2}}_{\mathsf{OAEP}_{\mathsf{s-clear}}, A, \mathsf{Ext}}(k) \quad \leq \quad & 3 \cdot \mathbf{Adv}^{\delta\text{-ext2}_\zeta}_{\mathcal{G}, \mathsf{OAEP}_{\mathsf{s-clear}}, A_G, \mathsf{Ext}_G}(k) + 5 \cdot \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G}, B_G}(k) \\ & + 2 \cdot \mathbf{Adv}^{\text{cr}}_{\mathcal{H}, B_H}(k) + 2p \cdot \mathbf{Adv}^{\text{xor-nm1}}_{\mathcal{F}, \mathcal{G}, B_F}(k) \end{aligned}$$

Using Lemmas 7.11 and 7.12,

$$\begin{aligned} \mathbf{Adv}^{\text{pa2}}_{\mathsf{OAEP}_{\mathsf{s-clear}}, A, \mathsf{Ext}}(k) \quad \leq \quad & 3 \cdot \mathbf{Adv}^{\eta\text{-ext2}_\zeta}_{\mathcal{G}, \mathcal{F}, A_G, \mathsf{Ext}_G}(k) + 9p \cdot \mathbf{Adv}^{\text{xor-ind2}_\zeta}_{\mathcal{F}, \mathcal{G}, C}(k) + 6p \cdot \mathbf{Adv}^{\text{prg}}_{\mathcal{G}, D}(k) \\ & + 12p \cdot \mathbf{Adv}^{\text{ext2}_\zeta}_{\mathcal{G}, C_G, A_G, \mathsf{Ext}'_G}(k) + 5 \cdot \mathbf{Adv}^{\text{n-cr}_\zeta}_{\mathcal{G}, B_G}(k) + 2 \cdot \mathbf{Adv}^{\text{cr}}_{\mathcal{H}, B_H}(k) + 2p \cdot \mathbf{Adv}^{\text{xor-nm1}}_{\mathcal{F}, \mathcal{G}, B_F}(k) \end{aligned}$$

Algorithm $B_{F,1}(f, K_G, x)$
$i \leftarrow 1$; $j \leftarrow\$ [p]$
$\mathbf{s} \leftarrow \varepsilon$; $\mathbf{c} \leftarrow \varepsilon$
$st \leftarrow \varepsilon$; $z \leftarrow \varepsilon$
$K_H \leftarrow\$ \mathcal{K}_H(1^k)$
$pk \leftarrow (K_G, K_H, f)$
$w \leftarrow\$ \mathsf{Coins}(k)$
$state \leftarrow (pk, w)$
Run $A^{\mathcal{D}\text{Sim}(\cdot),\text{EncSim}(\cdot)}(pk; w)$
Return (z, st)

Procedure $\mathcal{D}\text{Sim}(c)$
$(m, state) \leftarrow\$ \mathsf{Ext}(state, \mathbf{c}, c)$
Return m

Procedure $\text{EncSim}(\mathcal{M})$
$m \leftarrow \mathcal{M}(1^k, pk)$
If $i = j$ then
  $s \leftarrow m \| 0^\zeta \oplus x$; $z \leftarrow H(K_H, s)$
  $st \leftarrow (pk, s, w, j)$; Halt A
$\mathbf{c}[i] \leftarrow \mathsf{Enc}(pk, m)$
$\mathbf{s}[i] \leftarrow \mathbf{c}[i]|^{\mu+\zeta}$; $i \leftarrow i + 1$
Return $\mathbf{c}[i]$

Algorithm $B_{F,2}(st, y)$
$\mathsf{out}_1 \leftarrow \varepsilon$; $\omega \leftarrow \varepsilon$
$i \leftarrow 1$; $\mathbf{s} \leftarrow \varepsilon$; $\mathbf{c} \leftarrow \varepsilon$
$(pk, s, w, j) \leftarrow st$
$state \leftarrow (pk, w)$
Run $A^{\mathcal{D}\text{Sim}(\cdot),\text{EncSim}(\cdot)}(pk; w)$
Return (ω, y)

Procedure $\mathcal{D}\text{Sim}(c)$
$(s, y) \leftarrow c$
If $s \neq \mathbf{s}[j] \wedge s|_\zeta = \mathbf{s}[j]|_\zeta \wedge y \neq \mathbf{c}[j]|_\rho$
  $\mathsf{out}_1 \leftarrow y$
  $\omega \leftarrow H(K_H, s) \oplus H(K_H, \mathbf{s}[j])$
  Halt A
$(m, state) \leftarrow\$ \mathsf{Ext}(state, \mathbf{c}, c)$
Return m

Procedure $\text{EncSim}(\mathcal{M})$
$m \leftarrow \mathcal{M}(1^k, pk)$
$\mathbf{c}[i] \leftarrow \mathsf{Enc}(pk, m)$
If $i = j$ then
  $\mathbf{c}[i] \leftarrow (s, y)$
$\mathbf{s}[i] \leftarrow \mathbf{c}[i]|^{\mu+\zeta}$; $i \leftarrow i + 1$
Return $\mathbf{c}[i]$

Figure 60: **XOR-NM1 adversary B_F in the proof of Theorem 7.10.**

This completes the proof. ∎

Acknowledgments

We thank Ran Canetti, Eike Kiltz, and Adam Smith for insightful conversations. We also thank the audience in talks at ENS Paris and TU Darmstadt for helpful feedback. Adam O'Neill was partially supported by NSF grant CNS-1650419. Part of this work was carried out when he was a Mercator fellow at TU Darmstadt, and he thanks them for their hospitality. Mohammad Zaheri was supported by NSF grant No. 1565387 and NSF grant No. 1149832.

References

[1] P. Baecher, M. Fischlin, and D. Schröder. Expedient non-malleability notions for hash functions. In A. Kiayias, editor, *CT-RSA 2011*, volume 6558 of *LNCS*, pages 268–283, San Francisco, CA, USA, Feb. 14–18, 2011. Springer, Heidelberg, Germany. (Cited on page 8.)

[2] B. Barak. How to go beyond the black-box simulation barrier. In *42nd FOCS*, pages 106–115, Las Vegas, NV, USA, Oct. 14–17, 2001. IEEE Computer Society Press. (Cited on page 4.)

[3] G. Barthe, D. Pointcheval, and S. Zanella Béguelin. Verified security of redundancy-free encryption from rabin and rsa. In *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, CCS '12, pages 724–735, New York, NY, USA, 2012. ACM. (Cited on page 5, 13, 14, 19, 53, 54.)

[4] M. Bellare, A. Boldyreva, and A. O'Neill. Deterministic and efficiently searchable encryption. In A. Menezes, editor, *CRYPTO 2007*, volume 4622 of *LNCS*, pages 535–552, Santa Barbara, CA, USA, Aug. 19–23, 2007. Springer, Heidelberg, Germany. (Cited on page 30, 31.)

[5] M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway. Relations among notions of security for public-key encryption schemes. In H. Krawczyk, editor, *CRYPTO'98*, volume 1462 of *LNCS*, pages 26–45, Santa Barbara, CA, USA, Aug. 23–27, 1998. Springer, Heidelberg, Germany. (Cited on page 4, 5, 23.)

[6] M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway. Relations among notions of security for public-key encryption schemes. In *Proceedings of the 18th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '98, pages 26–45, London, UK, UK, 1998. Springer-Verlag. (Cited on page 9.)

[7] M. Bellare, V. T. Hoang, and S. Keelveedhi. Instantiating random oracles via UCEs. In R. Canetti and J. A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 398–415, Santa Barbara, CA, USA, Aug. 18–22, 2013. Springer, Heidelberg, Germany. (Cited on page 4, 6, 7, 8, 14, 25, 30, 32.)

[8] M. Bellare, V. T. Hoang, and S. Keelveedhi. Cryptography from compression functions: The UCE bridge to the ROM. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 169–187, Santa Barbara, CA, USA, Aug. 17–21, 2014. Springer, Heidelberg, Germany. (Cited on page 8.)

[9] M. Bellare, C. Namprempre, D. Pointcheval, and M. Semanko. The one-more-RSA-inversion problems and the security of Chaum's blind signature scheme. *Journal of Cryptology*, 16(3):185–215, June 2003. (Cited on page 4.)

[10] M. Bellare and A. Palacio. Towards plaintext-aware public-key encryption without random oracles. In P. J. Lee, editor, *ASIACRYPT 2004*, volume 3329 of *LNCS*, pages 48–62, Jeju Island, Korea, Dec. 5–9, 2004. Springer, Heidelberg, Germany. (Cited on page 4, 5, 9, 10, 23.)

[11] M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In V. Ashby, editor, *ACM CCS 93*, pages 62–73, Fairfax, Virginia, USA, Nov. 3–5, 1993. ACM Press. (Cited on page 3.)

[12] M. Bellare and P. Rogaway. Optimal asymmetric encryption. In A. D. Santis, editor, *EUROCRYPT'94*, volume 950 of *LNCS*, pages 92–111, Perugia, Italy, May 9–12, 1994. Springer, Heidelberg, Germany. (Cited on page 3, 4, 7, 9, 13.)

[13] M. Bellare and P. Rogaway. The exact security of digital signatures: How to sign with RSA and Rabin. In U. M. Maurer, editor, *EUROCRYPT'96*, volume 1070 of *LNCS*, pages 399–416, Saragossa, Spain, May 12–16, 1996. Springer, Heidelberg, Germany. (Cited on page 3.)

[14] M. Bellare and P. Rogaway. The security of triple encryption and a framework for code-based game-playing proofs. In S. Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 409–426, St. Petersburg, Russia, May 28 – June 1, 2006. Springer, Heidelberg, Germany. (Cited on page 16, 17, 21, 55, 56, 62.)

[15] N. Bitansky, R. Canetti, A. Chiesa, and E. Tromer. From extractable collision resistance to succinct non-interactive arguments of knowledge, and back again. In S. Goldwasser, editor, *ITCS 2012*, pages 326–349, Cambridge, MA, USA, Jan. 8–10, 2012. ACM. (Cited on page 8.)

[16] N. Bitansky, R. Canetti, O. Paneth, and A. Rosen. On the existence of extractable one-way functions. In D. B. Shmoys, editor, *46th ACM STOC*, pages 505–514, New York, NY, USA, May 31 – June 3, 2014. ACM Press. (Cited on page 25, 38.)

[17] N. Bitansky, R. Canetti, O. Paneth, and A. Rosen. On the existence of extractable one-way functions. *SIAM Journal on Computing*, 45(5):1910–1952, 2016. (Cited on page 6, 8.)

[18] D. Bleichenbacher. On the security of the kmov public key cryptosystem. In B. S. Kaliski, editor, *Advances in Cryptology — CRYPTO '97*, pages 235–248, Berlin, Heidelberg, 1997. Springer Berlin Heidelberg. (Cited on page 53.)

[19] J. Blömer and A. May. A tool kit for finding small roots of bivariate polynomials over the integers. In R. Cramer, editor, *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 251–267, Aarhus, Denmark, May 22–26, 2005. Springer, Heidelberg, Germany. (Cited on page 5.)

[20] A. Boldyreva, D. Cash, M. Fischlin, and B. Warinschi. Foundations of non-malleable hash and one-way functions. In M. Matsui, editor, *ASIACRYPT 2009*, volume 5912 of *LNCS*, pages 524–541, Tokyo, Japan, Dec. 6–10, 2009. Springer, Heidelberg, Germany. (Cited on page 8.)

[21] A. Boldyreva and M. Fischlin. Analysis of random oracle instantiation scenarios for OAEP and other practical schemes. In V. Shoup, editor, *CRYPTO 2005*, volume 3621 of *LNCS*, pages 412–429, Santa Barbara, CA, USA, Aug. 14–18, 2005. Springer, Heidelberg, Germany. (Cited on page 5, 8.)

[22] A. Boldyreva and M. Fischlin. On the security of OAEP. In X. Lai and K. Chen, editors, *ASIACRYPT 2006*, volume 4284 of *LNCS*, pages 210–225, Shanghai, China, Dec. 3–7, 2006. Springer, Heidelberg, Germany. (Cited on page 5, 7, 8, 11, 30, 32.)

[23] A. Boldyreva, H. Imai, and K. Kobara. How to strengthen the security of RSA-OAEP. *IEEE Trans. Information Theory*, 56(11):5876–5886, 2010. (Cited on page 7.)

[24] D. Boneh. Simplified OAEP for the RSA and Rabin functions. In J. Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 275–291, Santa Barbara, CA, USA, Aug. 19–23, 2001. Springer, Heidelberg, Germany. (Cited on page 19.)

[25] D. Boneh and G. Durfee. Cryptanalysis of rsa with private key d less than n0.292. In J. Stern, editor, *Advances in Cryptology — EUROCRYPT '99*, pages 1–11, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg. (Cited on page 53.)

[26] D. Boneh and M. K. Franklin. Identity based encryption from the Weil pairing. *SIAM Journal on Computing*, 32(3):586–615, 2003. (Cited on page 3.)

[27] D. R. L. Brown. A weak-randomizer attack on rsa-oaep with e = 3, 2005. (Cited on page 8.)

[28] C. Brzuska, P. Farshim, and A. Mittelbach. Indistinguishability obfuscation and UCEs: The case of computationally unpredictable sources. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 188–205, Santa Barbara, CA, USA, Aug. 17–21, 2014. Springer, Heidelberg, Germany. (Cited on page 4, 8.)

[29] R. Canetti. Towards realizing random oracles: Hash functions that hide all partial information. In B. S. Kaliski Jr., editor, *CRYPTO'97*, volume 1294 of *LNCS*, pages 455–469, Santa Barbara, CA, USA, Aug. 17–21, 1997. Springer, Heidelberg, Germany. (Cited on page 5, 8.)

[30] R. Canetti, Y. Chen, and L. Reyzin. On the correlation intractability of obfuscated pseudorandom functions. In *Theory of Cryptography - 13th International Conference, TCC*, pages 389–415, 2016. (Cited on page 4, 8.)

[31] R. Canetti and R. R. Dakdouk. Extractable perfectly one-way functions. In L. Aceto, I. Damgård, L. A. Goldberg, M. M. Halldórsson, A. Ingólfsdóttir, and I. Walukiewicz, editors, *ICALP 2008, Part II*, volume 5126 of *LNCS*, pages 449–460, Reykjavik, Iceland, July 7–11, 2008. Springer, Heidelberg, Germany. (Cited on page 6, 8, 23.)

[32] R. Canetti and R. R. Dakdouk. Towards a theory of extractable functions. In O. Reingold, editor, *TCC 2009*, volume 5444 of *LNCS*, pages 595–613. Springer, Heidelberg, Germany, Mar. 15–17, 2009. (Cited on page 6, 8, 23.)

[33] R. Canetti, O. Goldreich, and S. Halevi. The random oracle methodology, revisited. *J. ACM*, 51(4):557–594, 2004. (Cited on page 3, 4, 8.)

[34] R. Canetti, D. Micciancio, and O. Reingold. Perfectly one-way probabilistic hash functions (preliminary version). In *30th ACM STOC*, pages 131–140, Dallas, TX, USA, May 23–26, 1998. ACM Press. (Cited on page 8.)

[35] D. Coppersmith. Finding a small root of a univariate modular equation. In *Proceedings of the 15th Annual International Conference on Theory and Application of Cryptographic Techniques*, EUROCRYPT'96, pages 155–165, Berlin, Heidelberg, 1996. Springer-Verlag. (Cited on page 5, 14.)

[36] D. Coppersmith. Finding a small root of a univariate modular equation. In U. M. Maurer, editor, *EUROCRYPT'96*, volume 1070 of *LNCS*, pages 155–165, Saragossa, Spain, May 12–16, 1996. Springer, Heidelberg, Germany. (Cited on page 53.)

[37] J.-S. Coron, A. Kirichenko, and M. Tibouchi. A note on the bivariate Coppersmith theorem. *Journal of Cryptology*, 26(2):246–250, Apr. 2013. (Cited on page 5.)

[38] D. Dachman-Soled, R. Gennaro, H. Krawczyk, and T. Malkin. Computational extractors and pseudorandomness. In R. Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 383–403, Taormina, Sicily, Italy, Mar. 19–21, 2012. Springer, Heidelberg, Germany. (Cited on page 12, 19.)

[39] I. Damgård. Towards practical public key systems secure against chosen ciphertext attacks. In J. Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 445–456, Santa Barbara, CA, USA, Aug. 11–15, 1991. Springer, Heidelberg, Germany. (Cited on page 4.)

[40] Y. Dodis, I. Haitner, and A. Tentes. On the instantiability of hash-and-sign RSA signatures. In R. Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 112–132, Taormina, Sicily, Italy, Mar. 19–21, 2012. Springer, Heidelberg, Germany. (Cited on page 4.)

[41] Y. Dodis, R. Oliveira, and K. Pietrzak. On the generic insecurity of the full domain hash. In V. Shoup, editor, *CRYPTO 2005*, volume 3621 of *LNCS*, pages 449–466, Santa Barbara, CA, USA, Aug. 14–18, 2005. Springer, Heidelberg, Germany. (Cited on page 4.)

[42] Y. Dodis and A. Smith. Entropic security and the encryption of high entropy messages. In J. Kilian, editor, *TCC 2005*, volume 3378 of *LNCS*, pages 556–577, Cambridge, MA, USA, Feb. 10–12, 2005. Springer, Heidelberg, Germany. (Cited on page 30.)

[43] G. Durfee and P. Q. Nguyen. Cryptanalysis of the rsa schemes with short secret exponent from asiacrypt '99. In T. Okamoto, editor, *Advances in Cryptology — ASIACRYPT 2000*, pages 14–29, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg. (Cited on page 53.)

[44] M. Fischlin. Pseudorandom function tribe ensembles based on one-way permutations: Improvements and applications. In J. Stern, editor, *EUROCRYPT'99*, volume 1592 of *LNCS*, pages 432–445, Prague, Czech Republic, May 2–6, 1999. Springer, Heidelberg, Germany. (Cited on page 8.)

[45] E. Fujisaki, T. Okamoto, D. Pointcheval, and J. Stern. RSA-OAEP is secure under the RSA assumption. In J. Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 260–274, Santa Barbara, CA, USA, Aug. 19–23, 2001. Springer, Heidelberg, Germany. (Cited on page 11, 15.)

[46] E. Fujisaki, T. Okamoto, D. Pointcheval, and J. Stern. RSA-OAEP is secure under the RSA assumption. *Journal of Cryptology*, 17(2):81–104, Mar. 2004. (Cited on page 6, 7, 41.)

[47] B. Fuller, A. O'Neill, and L. Reyzin. A unified approach to deterministic encryption: New constructions and a connection to computational entropy. In R. Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 582–599, Taormina, Sicily, Italy, Mar. 19–21, 2012. Springer, Heidelberg, Germany. (Cited on page 12, 30, 31, 38, 58.)

[48] S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984. (Cited on page 9.)

[49] D. Gupta and A. Sahai. On constant-round concurrent zero-knowledge from a knowledge assumption. In *Progress in Cryptology—INDOCRYPT 2014*, pages 71–88, 2014. (Cited on page 23.)

[50] S. Halevi, S. Myers, and C. Rackoff. On seed-incompressible functions. In R. Canetti, editor, *TCC 2008*, volume 4948 of *LNCS*, pages 19–36, San Francisco, CA, USA, Mar. 19–21, 2008. Springer, Heidelberg, Germany. (Cited on page 8.)

[51] V. T. Hoang, J. Katz, A. O'Neill, and M. Zaheri. Selective-opening security in the presence of randomness failures. In J. H. Cheon and T. Takagi, editors, *Advances in Cryptology – ASIACRYPT 2016*, pages 278–306, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg. (Cited on page 3.)

[52] D. Hofheinz. All-but-many lossy trapdoor functions. In D. Pointcheval and T. Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 209–227, Cambridge, UK, Apr. 15–19, 2012. Springer, Heidelberg, Germany. (Cited on page 40.)

[53] D. Hofheinz, E. Kiltz, and V. Shoup. Practical chosen ciphertext secure encryption from factoring. *Journal of Cryptology*, 26(1):102–118, Jan. 2013. (Cited on page 42.)

[54] S. Hohenberger, A. Sahai, and B. Waters. Replacing a random oracle: Full domain hash from indistinguishability obfuscation. In P. Q. Nguyen and E. Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 201–220, Copenhagen, Denmark, May 11–15, 2014. Springer, Heidelberg, Germany. (Cited on page 4.)

[55] C. S. Jutla. On finding small solutions of modular multivariate polynomial equations. In K. Nyberg, editor, *Advances in Cryptology — EUROCRYPT'98*, pages 158–170, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg. (Cited on page 53.)

[56] E. Kiltz, P. Mohassel, and A. O'Neill. Adaptive trapdoor functions and chosen-ciphertext security. In H. Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 673–692, French Riviera, May 30 – June 3, 2010. Springer, Heidelberg, Germany. (Cited on page 7, 8.)

[57] E. Kiltz, A. O'Neill, and A. Smith. Instantiability of RSA-OAEP under chosen-plaintext attack. In T. Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 295–313, Santa Barbara, CA, USA, Aug. 15–19, 2010. Springer, Heidelberg, Germany. (Cited on page 5, 7, 14.)

[58] E. Kiltz and K. Pietrzak. On the security of padding-based encryption schemes - or - why we cannot prove OAEP secure in the standard model. In A. Joux, editor, *EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 389–406, Cologne, Germany, Apr. 26–30, 2009. Springer, Heidelberg, Germany. (Cited on page 4, 6, 8, 13.)

[59] E. Kiltz and K. Pietrzak. Personal communication, 2019. (Cited on page 6.)

[60] M. Lewko, A. O'Neill, and A. Smith. *Regularity of Lossy RSA on Subdomains and Its Applications*, pages 55–75. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. (Cited on page 8, 38.)

[61] A. C. S. G. H. L. A. R. Nir Bitansky, Ran Canetti and E. Tromer. The hunting of the SNARK. In *J. Cryptology*, volume 30, pages 989–1066, 2017. (Cited on page 6, 25.)

[62] P. Paillier and J. L. Villar. Trading one-wayness against chosen-ciphertext security in factoring-based encryption. In X. Lai and K. Chen, editors, *ASIACRYPT 2006*, volume 4284 of *LNCS*, pages 252–266, Shanghai, China, Dec. 3–7, 2006. Springer, Heidelberg, Germany. (Cited on page 8.)

[63] O. Pandey, R. Pass, and V. Vaikuntanathan. Adaptive one-way functions and applications. In D. Wagner, editor, *CRYPTO 2008*, volume 5157 of *LNCS*, pages 57–74, Santa Barbara, CA, USA, Aug. 17–21, 2008. Springer, Heidelberg, Germany. (Cited on page 4, 7, 12, 25.)

[64] C. Peikert and B. Waters. Lossy trapdoor functions and their applications. In R. E. Ladner and C. Dwork, editors, *40th ACM STOC*, pages 187–196, Victoria, British Columbia, Canada, May 17–20, 2008. ACM Press. (Cited on page 5, 8.)

[65] C. Peikert and B. Waters. Lossy trapdoor functions and their applications. *SIAM J. Comput.*, 40(6):1803–1844, 2011. (Cited on page 12.)

[66] C. Rackoff and D. R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In J. Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 433–444, Santa Barbara, CA, USA, Aug. 11–15, 1991. Springer, Heidelberg, Germany. (Cited on page 5.)

[67] C. Rackoff and D. R. Simon. Cryptographic defense against traffic analysis. In *25th ACM STOC*, pages 672–681, San Diego, CA, USA, May 16–18, 1993. ACM Press. (Cited on page 9.)

[68] R. L. Rivest, A. Shamir, and L. M. Adleman. A method for obtaining digital signature and public-key cryptosystems. *Communications of the Association for Computing Machinery*, 21(2):120–126, 1978. (Cited on page 11.)

[69] V. Shoup. OAEP reconsidered. In J. Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 239–259, Santa Barbara, CA, USA, Aug. 19–23, 2001. Springer, Heidelberg, Germany. (Cited on page 38.)

[70] V. Shoup. OAEP reconsidered. *Journal of Cryptology*, 15(4):223–249, 2002. (Cited on page 5, 6, 7, 41, 63.)

[71] A. Smith and Y. Zhang. *On the Regularity of Lossy RSA*, pages 609–628. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015. (Cited on page 8, 38.)

[72] M. Zhandry. The magic of ELFs. *J. Cryptology*, 32(3):825–866, 2019. (Cited on page 4.)

A Generalized SIE and CIE of RSA

In this section, we show black-box extractability properties of RSA, generalizing the work of Barthe *at al.* [3]. Namely, we show that RSA with small exponent is (i, j)-second input extractable and (i, j)-common input extractable for certain parameters i, j.

Coppersmith's technique. Our proofs rely on Coppersmith's technique [36] to find small integer roots of univariate and bivariate polynomials modulo N with unknown factorization. Let us state the results we use.

Proposition A.1 (Univariate Coppersmith) There is an algorithm that on inputs a monic integer polynomial $p(X)$ of degree δ with integer coefficients, and a positive integer N, outputting all integer solutions x_0 to $p(x_0) = 0 \bmod N$ with $|x_0| < N^{1/\delta}$ in time polynomial in $\log(N)$ and δ.

Proposition A.2 (Bivariate Coppersmith (Heuristic)) There is an algorithm that on inputs a polynomial $p(X, Y)$ of total degree δ with a monic monomial $X^a Y^{\delta-a}$ for some a, and a positive integer N, outputting all integer solutions x_0, y_0 to $p(x_0, y_0) = 0 \bmod N$ with $|x_0 y_0| < N^{1/\delta}$ in time polynomial in $\log(N)$ and δ.

Note while the bivariate Coppersmith algorithm is not know to provably run in polynomial-time, [55, 43, 25, 18] shows it works well in practice.

Main results. We now give the main results of this section. We use $t_{\mathrm{Cop}(N,\delta)}$ to denote the maximum running-time of the univariate and bivariate Coppersmith algorithms on inputs as above. We also use $t_{\mathrm{Euc}(N,\delta)}$ to denote the maximum running-time of the extended Euclidean algorithm on two univariate polynomials of at most degree δ over $\mathbb{Z}_N^*$.[7] Recall the RSA trapdoor permutation family, parameterized by N, e where $n = \lceil \log N \rceil$, is defined as $f_{N,e}(x) = x^e \bmod N$ for $x \in \mathbb{Z}_N^*$.

Theorem A.3 *The RSA trapdoor permutation family is (i, j)-second input extractable for $j - i > (1 - 1/e)n$. The extractor runs in time $t_{Cop(N,e)}$.*

Theorem A.4 *The RSA trapdoor permutation family is (i, j)-common input extractable for $j - i > (1 - 1/e^2)n$. The extractor runs in time $t_{Cop(N,e^2)} + t_{Euc(N,e)}$.*

Proofs of main results. We now give the proofs of the main results.

Proof: (**of Theorem A.3**)

Firstly, let's recall the definition of (i, j)-second input extractable. Let $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a trapdoor permutation family with domain TDom. For $i, j \in \mathbb{N}$, we say $\mathcal{F}$ is (i, j)-*second input extractable* if there exists an efficient extractor $\mathcal{E}$ such that for every $f \in [\mathsf{Kg}(1^k)]$ and every $x \in \mathsf{TDom}(k)$, extractor $\mathcal{E}$ on inputs $f, f(x), x|_{i+1}^{j}$ outputs x.

For any element $x \in \mathbb{Z}_N^*$ and $i, j \in [0, n]$, $i < j$, x can be uniquely represented as $x = s \cdot 2^j + r \cdot 2^i + t$, where $s \in \{0,1\}^{n-j}$, $r \in \{0,1\}^{j-i}$, and $t \in \{0,1\}^i$. Notice that if $j = n$ or $i = 0$, we will remove s or t from the formula respectively. Now, we can rewrite RSA as a function of three arguments:

$$f_{N,e}(x) = f_{N,e}(s, r, t) = (s \cdot 2^j + r \cdot 2^i + t)^e \bmod N \ .$$

The high level idea for (i, j)-*second input extractable* is to solve the monic integer polynomial through coppersmith algorithm. Specifically, we will construct the extractor $\mathcal{E}$ in several cases:

[7] Although Z_N^* is not a field, if the algorithm fails it can recover a non-trivial factor of N.

- $i = 0$. Then the item t will be removed from RSA function, we can represent it as:

$$f_{N,e}(x) = f_{N,e}(s, r) = (s \cdot 2^j + r)^e \bmod N \ .$$

To construct an extractor $\mathcal{E}$ on inputs $r = x|_{i+1}^{j}$ and $c = f_{N,e}(x)$, we can consider the polynomial $p(X) = 0 \bmod N$ for $p(X) = (X \cdot 2^j + r)^e - c$. The Coppersmith univariate algorithm requires monic polynomial to find the root s. However, $p(X)$ is not monic polynomial. Notice that j and e are public, so we can easily find the inverse of $2^{je} \in \mathbb{Z}_N^*$, and multiply $p(X)$ to get a new monic polynomial. On the other hand, the Coppersmith algorithm can find only roots $s < N^{1/e}$, which means $2^{n-j} < N^{1/e}$, or equivalently, $j > (1 - 1/e)n$. The running time of extractor $\mathcal{E}$ can be bounded by the running time of Coppersmith algorithm $t_{\mathrm{Cop}(N,e)}$.

- $j = n$. This work has been shown in section 5.1 of [3]. The requirement for i is $i < n/e$ and the extractor $\mathcal{E}$ runs within time $t_{\mathrm{Cop}(N,e)}$.

- $i > 0$ AND $j < n$. This case will be slightly different from the first case. The extractor $\mathcal{E}$ on inputs $r = x|_{i+1}^{j}$ and $c = f_{N,e}(x)$ outputs s, t such that $f_{N,e}(s, r, t) = c$. By using the same strategy, we construct polynomial $p(X, Y) = (X \cdot 2^j + r \cdot 2^i + Y)^e - c \bmod N$ with two variables X and Y. The bivariate Coppersmith algorithm could find all integer solutions x_0, y_0 such that $|x_0 y_0| < N^{1/e}$, which equals to $2^{n-j} \cdot 2^i < N^{1/e}$, or in other words, such that $j - i > (1 - 1/e)n$. The extractor $\mathcal{E}$ executes within time $t_{\mathrm{Cop}(N,e)}$.

Combining these 3 cases, we thus construct an efficient (i, j)-*second input extractable* algorithm $\mathcal{E}$ running within time $t_{\mathrm{Cop}(N,e)}$ when $j - i > (1 - 1/e)n$. ▌

Proof: (**of Theorem A.4**)

Again, let's recall the definition of (i, j)-*common input extractable*. Let $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a trapdoor permutation family with domain TDom. For $i, j \in \mathbb{N}$, we say $\mathcal{F}$ is (i, j)-*common input extractable* if there exists an efficient extractor $\mathcal{E}$ such that for every $f \in [\mathsf{Kg}(1^k)]$ and every $x_1, x_2 \in \mathsf{TDom}(k)$, extractor $\mathcal{E}$ on inputs $f, f(x_1), f(x_2)$ outputs x_1, x_2 if $x_1|_{i+1}^{j} = x_2|_{i+1}^{j}$.

Given two different $c_1 = f(x_1), c_2 = f(x_2)$, our goal is to find s_1, r, t_1 and s_2, r, t_2 such that $c_1 = (s_1 \cdot 2^j + r \cdot 2^i + t_1)^e \bmod N$ and $c_2 = (s_2 \cdot 2^j + r \cdot 2^i + t_2)^e \bmod N$. Let us consider several cases:

- $i = 0$. In this case, t_1 and t_2 will be removed in the formula. Consider following two polynomials

$$\begin{aligned} p_1(X, Y) &= X^e - c_1 \bmod N \\ p_2(X, Y) &= (X + Y \cdot 2^j)^e - c_2 \bmod N \end{aligned}$$

When $x_0 = s_1 \cdot 2^j + r$ and $y_0 = s_2 - s_1$, both polynomials evaluate to 0. Taking $p_1(X, Y)$ and $p_2(X, Y)$ as one variable polynomial over X, the determinant of the $2e \times 2e$ Sylvester Matrix is a polynomial in Y. On the other hand, the resultant $Res(p_1, p_2, X)$, which equals to the determinant of the Sylvester Matrix, has root at point $Y = y_0$ since at point $Y = y_0$, $p_1(X, y_0)$ and $p_2(X, y_0)$ will share the same root x_0. Therefore, once we get get $Res(p_1, p_2, X)$ by computing Sylvester Matrix, we can use univariate Coppersmith algorithm solve polynomial $Res(p_1, p_2, X)$. Notice the specific form of the Sylvester Matrix, a straightforward but tedious calculation shows that the degree of $Res(p_1, p_2, X)$ is e^2 and the coefficient of Y^{e^2} is 2^{je^2}. We can easily adjust the coefficient of Y^{e^2} to 1 by multiplying the inverse of $2^{je^2} \in \mathbb{Z}_N^*$. The univariate Coppersmith algorithm requires $|y_0| < N^{1/e^2}$, or equivalently, $j > (1 - 1/e^2)n$. Once we work out y_0, $p_1(X, y_0)$ and $p_2(X, y_0)$ share the same and unique root x_0. Hence, $x - x_0$(or power of $(x - x_0)$) is a common factor of these two polynomials and can be found by extended Euclidean algorithm. The running time of extractor $\mathcal{E}$ could be bounded by the running time of Coppersmith algorithm $t_{\mathrm{Cop}(N,e^2)}$ and the running time of extended Euclidean algorithm $t_{\mathrm{Euc}(N,e)}$.

- $j = n$. This work has also been shown in section 5.1 of [3]. The requirement for i is $i < n/e^2$ and the extractor $\mathcal{E}$ runs within time $t_{C(N,e^2)} + t_{\mathrm{Euc}(N,e)}$.

- $i > 0$ AND $j < n$. The high level idea is almost the same as the first case, while the detail differs. Consider following two polynomials:

$$\begin{aligned} p_1(X, Y_1.Y_2) &= X^e - c_1 \bmod N \\ p_2(X, Y_1, Y_2) &= (X + Y_1 \cdot 2^j + Y_2)^e - c_2 \bmod N \end{aligned}$$

  Both polynomials should be equal to 0 at point $(x_0 = s_1 \cdot 2^j + r \cdot 2^i + t_1, y_1 = s_2 - s_1, y_2 = t_2 - t_1)$. Hence, the resultant polynomial $Res(p_1, p_2, X)$ over X has roots y_1 and y_2, since $p_1(X, y_1.y_2)$ and $p_2(X, y_1.y_2)$ share the same root x_0. On the other hand, The determinant of the $2e \times 2e$ Sylvester Matrix associated to the polynomial p_1 and p_2 over X, which equal to the resultant polynomial $Res(p_1, p_2, X)$, is a polynomial with total degree e^2 and has one monic monomial $Y_2^{e^2}$. Therefore, we can use bivariate Coppersmith algorithm get the roots y_1 and y_2 for polynomial $Res(p_1, p_2, X)$. Notice that bivariate Coppersmith algorithm requires $|y_1 y_2| < N^{1/e^2}$, which implies $j - i > (1 - 1/e^2)n$. The following part, including solving x_0 and running time will be same as the first case.

In summary, we have an efficient (i, j)-*common input extractable* algorithm for RSA if $j - i > (1 - 1/e^2)n$, as required. ∎

B IND-CPA Result Under Partial One-Wayness

Theorem B.1 Let n, μ, ζ, ρ be integer parameters. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ be a pseudorandom generator and $\mathcal{H} : \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be a RO. Let $\mathcal{F}$ be a family of trapdoor permutations with domain $\{0,1\}^n$, where $n = \mu + \zeta + \rho$. Suppose $\mathcal{F}$ is $(\mu + \zeta)$-partial one-way. Then $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ is IND-CPA. In particular, for any adversary $A = (A_1, A_2)$, there are an adversary D and an inverter I such that,

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}[\mathcal{G},\mathcal{H},\mathcal{F}],A}(k) \le 2q \cdot \mathbf{Adv}^{\text{pow}}_{\mathcal{F},I}(k) + 6 \cdot \mathbf{Adv}^{\text{prg}}_{\mathcal{G},D}(k) + \frac{2q}{2^{\mu+\zeta}} \ .$$

where q is the total number of random-oracle queries of A. The running time of D and I are about that of A.

Proof: Consider games G_1–G_6 in Figures 61–62. Each game maintains two independent random oracles RO and $\overline{\text{RO}}$. Procedure RO maintains a local array H as follows:

Procedure RO(v)
If $H[v] = \perp$ then $H[v] \leftarrow_{\$} \{0,1\}^\rho$
Return $H[v]$

For simplicity, we omit the code of RO, $\overline{\text{RO}}$ in the games. In each game, we use RO_1 to denote the oracle interface of adversary A_1 and message samplers $\mathcal{M}_0, \mathcal{M}_1$ and we use RO_2 to denote the oracle interface of adversary A_2. Game G_1 corresponds to game $\text{IND-CPA}^A_{\mathsf{OAEP}^n}$. Then

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}^n,A}(k) \le 2 \cdot \Pr[G_1(k) \Rightarrow 1] - 1 \ .$$

We now explain the game chain. Game G_2 is identical to game G_1, except in the encryption of message m_b, if either adversary A_1 or message sampler $\mathcal{M}_b$ queried s to their random oracle RO_1, then it chooses a fresh random value for $H[s]$. Games G_1 and G_2 are identical-until-bad_1, and thus from the Fundamental Lemma of Game-playing [14],

$$\Pr\left[\, G_1(k) \Rightarrow 1 \,\right] - \Pr\left[\, G_2(k) \Rightarrow 1 \,\right] \le \Pr\left[\, G_2(k) \text{ sets } \mathsf{bad}_1 \,\right] \ .$$

Now, consider adversary B attacking the psudorandom genetator G in Figure 63. We know that, $\mathbf{Adv}^{\text{prg}}_{G,B}(k) = 2 \cdot \Pr\left[\, \text{PRG-DIST}^B_G(k) \Rightarrow 1 \,\right] - 1$. Let PRG-REAL^B_G be the game identical to game PRG-DIST^B_G condition on $b = 1$, and PRG-RAND^B_G be the game identical to game PRG-DIST^B_G condition on $b = 0$. Then,

$$\mathbf{Adv}^{\text{prg}}_{G,B}(k) = \Pr\left[\, \text{PRG-REAL}^B_G \Rightarrow 1 \,\right] - \Pr\left[\, \text{PRG-RAND}^B_G \Rightarrow 1 \,\right] \ .$$

Games $G_1(k)$, $G_2(k)$	**Games** $G_3(k)$, $G_4(k)$
$b \leftarrow\!\!\$\ \{0,1\}$; $K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$	$b \leftarrow\!\!\$\ \{0,1\}$; $K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_G, f)$	$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_G, f)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{RO}_1(\cdot)}(1^k, pk)$	$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{RO}_1(\cdot)}(1^k, pk)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{RO}_1(\cdot)}(1^k, pk)$	$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{RO}_1(\cdot)}(1^k, pk)$
$r \leftarrow\!\!\$\ \{0,1\}^\rho$; $x \leftarrow G(K_G, r)$	$r \leftarrow\!\!\$\ \{0,1\}^\rho$; $x \leftarrow G(K_G, r)$
$s \leftarrow x \oplus (m_b \| 0^\zeta)$	$s \leftarrow x \oplus (m_b \| 0^\zeta)$; $t \leftarrow\!\!\$\ \{0,1\}^\rho$
If $H[s] \neq \bot$ then	$z \leftarrow t \oplus r$; $H[s] \leftarrow z$; $c \leftarrow f(s \| t)$
$\quad \mathsf{bad}_1 \leftarrow \mathsf{true}$; $H[s] \leftarrow\!\!\$\ \{0,1\}^\rho$	$d \leftarrow\!\!\$\ A_2^{\mathrm{RO}_2(\cdot)}(c, state)$
Else $H[s] \leftarrow\!\!\$\ \{0,1\}^\rho$	Return $(b = d)$
$z \leftarrow H[s]$; $t \leftarrow z \oplus r$; $c \leftarrow f(s \| t)$	**Procedure** $\mathrm{RO}_1(v)$
$d \leftarrow\!\!\$\ A_2^{\mathrm{RO}_2(\cdot)}(c, state)$	Return $\mathrm{RO}(v)$
Return $(b = d)$	**Procedure** $\mathrm{RO}_2(v)$
Procedure $\mathrm{RO}_1(v)$	If $v = s$ then
Return $\mathrm{RO}(v)$	$\quad \mathsf{bad}_2 \leftarrow \mathsf{true}$; return $\overline{\mathrm{RO}}(v)$
Procedure $\mathrm{RO}_2(v)$	Return $\mathrm{RO}(v)$
Return $\mathrm{RO}(v)$	

Figure 61: **Games G_1–G_4 in the proof of Theorem B.1.**

Games $G_5(k)$	**Games** $G_6(k)$
$b \leftarrow\!\!\$\ \{0,1\}$; $K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$	$b \leftarrow\!\!\$\ \{0,1\}$; $K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_G, f)$	$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_G, f)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{RO}_1(\cdot)}(1^k, pk)$	$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{RO}_1(\cdot)}(1^k, pk)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{RO}_1(\cdot)}(1^k, pk)$	$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{RO}_1(\cdot)}(1^k, pk)$
$r \leftarrow\!\!\$\ \{0,1\}^\rho$; $x \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$	$r \leftarrow\!\!\$\ \{0,1\}^\rho$; $s \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$
$s \leftarrow x \oplus (m_b \| 0^\zeta)$	$x \leftarrow s \oplus (m_b \| 0^\zeta)$
$t \leftarrow\!\!\$\ \{0,1\}^\rho$; $c \leftarrow f(s \| t)$	$t \leftarrow\!\!\$\ \{0,1\}^\rho$; $c \leftarrow f(s \| t)$
$d \leftarrow\!\!\$\ A_2^{\mathrm{RO}_2(\cdot)}(c, state)$	$d \leftarrow\!\!\$\ A_2^{\mathrm{RO}_2(\cdot)}(c, state)$
Return $(b = d)$	Return $(b = d)$
Procedure $\mathrm{RO}_1(v)$	**Procedure** $\mathrm{RO}_1(v)$
Return $\mathrm{RO}(v)$	Return $\mathrm{RO}(v)$
Procedure $\mathrm{RO}_2(v)$	**Procedure** $\mathrm{RO}_2(v)$
If $v = s$ then return $\overline{\mathrm{RO}}(v)$	If $v = s$ then return $\overline{\mathrm{RO}}(v)$
Return $\mathrm{RO}(v)$	Return $\mathrm{RO}(v)$

Figure 62: **Games G_5, G_6 in the proof of Theorem B.1.**

Note that $\Pr\left[\,\text{PRG-REAL}_G^B \Rightarrow 1\,\right] = \Pr\left[\,G_2(k) \text{ sets } \mathsf{bad}_1\,\right]$. Moreover, in the PRG-RAND_G^B, the probability adversary A queries for s is uniformly random. Multiplying for q random-oracle queries we have $\Pr[\text{PRG-RAND}_G^B \Rightarrow 1] \leq q/2^{\mu+\zeta}$. Thus

$$\Pr\left[\,G_2(k) \text{ sets } \mathsf{bad}_1\,\right] \leq \mathbf{Adv}^{\mathrm{prg}}_{G,B}(k) + \frac{q}{2^{\mu+\zeta}} \ .$$

In game G_3, we reorder the code of game G_2 in producing t. The change is conservative, meaning that $\Pr[G_2(k) \Rightarrow 1] = \Pr[G_3(k) \Rightarrow 1]$. Game G_4 is identical to game G_3, except in procedure RO_2, if adversary A_2 make a query for s, then the oracle lies, calling $\overline{\mathrm{RO}}$ instead. Game G_3 and game G_4 are identical-until-bad_2, and based on Fundamental Lemma of Game-playing [14],

$$\Pr\left[\,G_3(k) \Rightarrow 1\,\right] - \Pr\left[\,G_4(k) \Rightarrow 1\,\right] \leq \Pr\left[\,G_4(k) \text{ sets } \mathsf{bad}_2\,\right] \ .$$

Consider adversary C attacking the psudorandom genetator G in Figure 64. Let PRG-REAL_G^C be the game identical to game PRG-DIST_G^C condition on $b = 1$, and PRG-RAND_G^C be the game identical to game PRG-DIST_G^C

Algorithm $B(K_G, x)$	**Procedure** $\text{ROSIM}_1(v)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $\mathsf{out} \leftarrow 0$	If $H[v] = \perp$ then
$pk \leftarrow (K_G, f)$; $b \leftarrow\!\!\$\ \{0,1\}$	$H[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	Return $H[v]$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	
$s \leftarrow x \oplus (m_b \| 0^\zeta)$	
If $H[s] \neq \perp$ then $\mathsf{out} \leftarrow 1$	
Return out	

Figure 63: **Adversary B in the proof of Theorem B.1.**

Algorithm $C(K_G, x)$	**Procedure** $\text{ROSIM}_1(v)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $\mathsf{out} \leftarrow 0$	If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
$pk \leftarrow (K_G, f)$; $b \leftarrow\!\!\$\ \{0,1\}$	Return $H[v]$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	**Procedure** $\text{ROSIM}_2(v)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	If $v = s$ then
$s \leftarrow x \oplus (m_b \| 0^\zeta)$; $t \leftarrow\!\!\$\ \{0,1\}^\rho$	$\mathsf{out} \leftarrow 1$; Halt run of A_2
$c \leftarrow f(s \| t)$	If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$\ \{0,1\}^\rho$
Run $A_2^{\text{ROSIM}_2(\cdot)}(c, state)$	Return $H[v]$
Return out	

Figure 64: **Adversary C in the proof of Theorem B.1.**

condition on $b = 0$. Then,

$$\mathbf{Adv}^{\text{prg}}_{G,C}(k) = \Pr\left[\,\text{PRG-REAL}^C_G \Rightarrow 1\,\right] - \Pr\left[\,\text{PRG-RAND}^C_G \Rightarrow 1\,\right] \ .$$

Note that $\Pr\left[\,\text{PRG-REAL}^C_G \Rightarrow 1\,\right] = \Pr\left[\,G_4(k) \text{ sets } \mathsf{bad}_2\,\right]$. To bound the probability of game PRG-RAND^C_G outputs 1, we construct inverter I attacking the family of partial one-way trapdoor permutation $\mathcal{F}$ in Figure 65. Note that if adversary A_2 queries for s then inverter I could invert challenge c. Hence, we have $\Pr\left[\,\text{PRG-RAND}^C_G \Rightarrow 1\,\right] \leq q \cdot \mathbf{Adv}^{\text{pow}}_{\mathcal{F},I}(k)$. Thus,

$$\Pr\left[\,G_4(k) \text{ sets } \mathsf{bad}_2\,\right] \leq \mathbf{Adv}^{\text{prg}}_{G,C}(k) + q \cdot \mathbf{Adv}^{\text{pow}}_{\mathcal{F},I}(k) \ .$$

Next, game G_5 is identical to game G_4, except we are using completely random x in the encryption phase instead of using the psudorandom value $G(K, r)$. Consider adversary D as shown in Figure 66. Then

$$\Pr[G_4(k) \Rightarrow 1] - \Pr[G_5(k) \Rightarrow 1] \leq \mathbf{Adv}^{\text{prg}}_{G,D}(k) \ .$$

In game G_6, we reorder the code of game G_5 in producing s. The change is conservative, meaning that $\Pr[G_5(k) \Rightarrow 1] = \Pr[G_6(k) \Rightarrow 1]$. Note that, $\Pr[G_6(k) \Rightarrow 1] = 1/2$, since the distribution of ciphertexts is completely independent of bit b. Assuming that the advantage of adversary D is greater than the advantage of adversaries B and C, we have

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}^n,A}(k) \leq 2q \cdot \mathbf{Adv}^{\text{pow}}_{\mathcal{F},I}(k) + 6 \cdot \mathbf{Adv}^{\text{prg}}_{G,D}(k) + \frac{2q}{2^{\mu+\zeta}} \ .$$

This completes the proof. ∎

C $IND-CPA-KI implies $SIM-CPA-KI

Theorem C.1 Let $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE scheme. Let A be a $SIM-CPA-KI adversary against PKE with respect to message sampler $\mathcal{M}$. Then, there exist a simulator S and an $IND-CPA-KI adversary B with respect to message samplers $\mathcal{M}_0, \mathcal{M}_1$ such that for all $k \in \mathbb{N}$

$$\mathbf{Adv}^{\text{\$sim-cpa-ki}}_{\mathsf{PKE},A,S,\mathcal{M}}(k) \leq 81 \cdot \mathbf{Adv}^{\text{\$ind-cpa-ki}}_{\mathsf{PKE},B,\mathcal{M}_0,\mathcal{M}_1}(k) + \left(\frac{3}{4}\right)^k \ .$$

Algorithm $I(f, c)$	**Procedure** $\text{ROSIM}_1(v)$
$b \leftarrow\!\!\$ \{0,1\}$; $j \leftarrow 0$; $\mathsf{out} \leftarrow \perp$; $i \leftarrow\!\!\$ [q]$	If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$ \{0,1\}^\rho$
$K_G \leftarrow\!\!\$ \mathcal{K}_G(1^k)$; $pk \leftarrow (K_G, f)$	Return $H[v]$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$ A_1^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	**Procedure** $\text{ROSIM}_2(v)$
$m_b \leftarrow\!\!\$ \mathcal{M}_b^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	$j \leftarrow j + 1$
Run $A_2^{\text{ROSIM}_2(\cdot)}(c, state)$	If $j = i$ then
Return out	$\quad \mathsf{out} \leftarrow v$; Halt run of A_2
	If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$ \{0,1\}^\rho$
	Return $H[v]$

Figure 65: **Inverter I in the proof of Theorem B.1.**

Algorithm $D(K_G, x)$	**Procedure** $\text{ROSIM}_1(v)$
$(f, f^{-1}) \leftarrow\!\!\$ \mathsf{Kg}(1^k)$	If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$ \{0,1\}^\rho$
$pk \leftarrow (K_G, f)$; $b \leftarrow\!\!\$ \{0,1\}$	Return $H[v]$
$(\mathcal{M}_0, \mathcal{M}_1, \text{state}) \leftarrow\!\!\$ A_1^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	**Procedure** $\text{ROSIM}_2(v)$
$m_b \leftarrow\!\!\$ \mathcal{M}_b^{\text{ROSIM}_1(\cdot)}(1^k, pk)$	If $v = s$ then $H[v] \leftarrow\!\!\$ \{0,1\}^\rho$
$s \leftarrow x \oplus (m_b \| 0^\zeta)$; $t \leftarrow\!\!\$ \{0,1\}^\rho$	If $H[v] = \perp$ then $H[v] \leftarrow\!\!\$ \{0,1\}^\rho$
$c \leftarrow f(s \| t)$	Return $H[v]$
$b' \leftarrow\!\!\$ A_2^{\text{ROSIM}_2(\cdot)}(c, state)$	
Return $(b = b')$	

Figure 66: **Adversary D in the proof of Theorem B.1.**

where $\mathcal{M}_0, \mathcal{M}_1$ are 2-induced distributions of $\mathcal{M}$.

Proof: The proof is similar to the proof of Theorem 3.1 from [47]. We begin by showing that it is suffices to consider \$SIM-CPA-KI adversaries where the output of A.f is boolean.

Claim C.2 Let $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a R-PKE scheme. Let A be a \$SIM-CPA-KI adversary against PKE with respect to message sampler $\mathcal{M}$. Then, there is a boolean \$SIM-CPA-KI adversary B such that for all $k \in \mathbb{N}$

$$\mathbf{Adv}^{\text{\$sim-cpa-ki}}_{\mathsf{PKE},A,S_A,\mathcal{M}}(k) \leq 2\mathbf{Adv}^{\text{\$sim-cpa-ki}}_{\mathsf{PKE},B,S_B,\mathcal{M}}(k) \ .$$

where the running time of B is about that of A plus $\mathcal{O}(\ell)$.

Proof of Claim C.2. Let B be the adversary attacking PKE and S_B be the corresponding simulator as specified in Figure 67. Let E_A and E_B be the events that games \$SIM-CPA-KI-REAL$^{A,\mathcal{M}}_{\mathsf{PKE}}$ and \$SIM-CPA-KI-REAL$^{B,\mathcal{M}}_{\mathsf{PKE}}$ output 1, respectively. Hence,

$$\begin{aligned} \Pr[E_B] &= \Pr[E_A] + \frac{1}{2}(1 - \Pr[E_A]) \\ &= \frac{1}{2}\Pr[E_A] + \frac{1}{2} \ . \end{aligned}$$

Let T_A and T_B be the events that games \$SIM-CPA-KI-IDEAL$^{A,S_A,\mathcal{M}}_{\mathsf{PKE}}$ and \$SIM-CPA-KI-IDEAL$^{B,S_B,\mathcal{M}}_{\mathsf{PKE}}$ output 1, respectively. Similarly, we have $\Pr[T_B] = \Pr[T_A]/2 + 1/2$. Thus, $\mathbf{Adv}^{\text{\$sim-cpa-ki}}_{\mathsf{PKE},A,S_A,\mathcal{M}}(k) \leq 2 \cdot \mathbf{Adv}^{\text{\$sim-cpa-ki}}_{\mathsf{PKE},B,S_B,\mathcal{M}}(k)$. This completes the proof of Claim C.2.

Next, we claim that it is suffices to consider balanced \$SIM-CPA-KI adversaries meaning the probability the partial information is 1 or 0 is approximately 1/2. We call A *δ-balanced* boolean \$SIM-CPA-KI adversary if for all $b \in \{0,1\}$

$$\left| \Pr[t = b \ : \ t \leftarrow\!\!\$ A.\text{f}(m, param)] - \frac{1}{2} \right| \leq \delta \ .$$

for all *param* and m output by A.pg and $\mathcal{M}$, respectively.

Algorithm $B.\mathrm{pg}(1^k)$	**Algorithm** $B.\mathrm{f}(m, pars)$
$param \leftarrow\!\!\$\ A.\mathrm{pg}(1^k)$	$(r, param) \leftarrow pars$
$r \leftarrow\!\!\$\ \{0,1\}^{\mathsf{A.f.rl}(k)}$	$t \leftarrow\!\!\$\ A.\mathrm{f}(m, param)$
$pars \leftarrow (r, param)$	Return $\langle r, t\rangle$
Return $pars$	**Simulator** $S_B(pk, \lvert m\rvert, pars)$
Algorithm $B.\mathrm{g}(pk, c, pars)$	$(r, param) \leftarrow pars$
$(r, param) \leftarrow pars$	$\omega \leftarrow\!\!\$\ S_A(pk, \lvert m\rvert, param)$
$\omega \leftarrow\!\!\$\ A.\mathrm{g}(pk, c, param)$	Return $\langle r, \omega\rangle$
Return $\langle r, \omega\rangle$	

Figure 67: **$SIM-CPA-KI adversary B and simulator S_B in the proof of Theorem C.1.**

Algorithm $C.\mathrm{pg}(1^k)$	**Algorithm** $C.\mathrm{f}(m, param)$
$param \leftarrow\!\!\$\ B.\mathrm{pg}(1^k)$	$t \leftarrow\!\!\$\ B.\mathrm{f}(m, param)$
Return $param$	$j \leftarrow\!\!\$\ \{1, \cdots 2(1/\delta)+1\}$
Algorithm $C.\mathrm{g}(pk, c, param)$	If $j \le 1/\delta$ then return 0
$\omega \leftarrow\!\!\$\ B.\mathrm{g}(pk, c, param)$	Else if $j \le 2(1/\delta)$ return 1
$i \leftarrow\!\!\$\ \{1, \cdots 2(1/\delta)+1\}$	Else return t
If $i \le 1/\delta$ then return 0	**Simulator** $S_C(pk, \lvert m\rvert, param)$
Else if $i \le 2(1/\delta)$ return 1	$\omega \leftarrow\!\!\$\ S_B(pk, \lvert m\rvert, param)$
Else return ω	$k \leftarrow\!\!\$\ \{1, \cdots 2(1/\delta)+1\}$
	If $k \le 1/\delta$ then return 0
	Else if $k \le 2(1/\delta)$ return 1
	Else return ω

Figure 68: **$SIM-CPA-KI adversary C and simulator S_C in the proof of Theorem C.1.**

Claim C.3 Let $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a R-PKE scheme. Let B be a boolean $SIM-CPA-KI adversary against PKE with respect to the message sampler $\mathcal{M}$. Then for any $0 \le \delta < 1/2$, there is a δ-*balanced* boolean $SIM-CPA-KI adversary C such that for all $k \in \mathbb{N}$

$$\mathbf{Adv}^{\$\text{sim-cpa-ki}}_{\mathsf{PKE},B,S_B,\mathcal{M}}(k) \le \left(\frac{2}{\delta}+1\right)^2 \cdot \mathbf{Adv}^{\$\text{sim-cpa-ki}}_{\mathsf{PKE},C,S_C,\mathcal{M}}(k) \ .$$

where the running time of C is about that of B plus $\mathcal{O}(1/\delta)$

Proof of Claim C.3. For simplicity, we assume $1/\delta$ is an integer. Let C be the adversary attacking PKE and S_C be the corresponding simulator as specified in Figure 68. It is easy to see that C is δ-*balanced*, since for all $b \in \{0,1\}$

$$\left|\Pr\left[\, t = b \ : \ t \leftarrow\!\!\$\ C.\mathrm{f}(m, param)\,\right] - \frac{1}{2}\right| \le \frac{1}{2/\delta+1} \ .$$

Let E_B and E_C be the events that games $SIM-CPA-KI-REAL$^{B,\mathcal{M}}_{\mathsf{PKE}}$ and $SIM-CPA-KI-REAL$^{C,\mathcal{M}}_{\mathsf{PKE}}$ output 1, respectively. Let T be the event that $i, j = 2/\delta + 1$. Thus,

$$\begin{aligned}\Pr[\,E_C\,] &= \Pr[\,E_C \mid T\,]\cdot\Pr[\,T\,] + \Pr\left[\,E_C \mid \overline{T}\,\right]\cdot\Pr\left[\,\overline{T}\,\right] \\ &= \left(\frac{1}{2/\delta+1}\right)^2 \Pr[\,E_B\,] + \frac{1}{2}\Pr\left[\,\overline{T}\,\right] \ .\end{aligned}$$

Let T_C and T_B be the events that games $SIM-CPA-KI-IDEAL$^{C,S_C,\mathcal{M}}_{\mathsf{PKE}}$and $SIM-CPA-KI-IDEAL$^{B,S_B,\mathcal{M}}_{\mathsf{PKE}}$ output 1, respectively. Similarly, we have

$$\Pr[\,T_C\,] = \left(\frac{1}{2/\delta+1}\right)^2 \Pr[\,T_B\,] + \frac{1}{2}\Pr\left[\,\overline{T}\,\right] \ .$$

Algorithm $D.\mathrm{pg}(1^k)$	**Algorithm** $D.\mathrm{g}(pk, c, param)$
$param \leftarrow\!\!\$\ C.\mathrm{pg}(1^k)$	$b' \leftarrow\!\!\$\ C.\mathrm{g}(pk, c, param)$
Return $param$	Return b'
Algorithm $\mathcal{M}_0(1^k, param)$	**Algorithm** $\mathcal{M}_1(1^k, param)$
For $i = 1$ to $k/2$ do:	For $i = 1$ to $k/2$ do:
$\quad m \leftarrow\!\!\$\ \mathcal{M}(1^k, param)$	$\quad m \leftarrow\!\!\$\ \mathcal{M}(1^k, param)$
$\quad$ If $C.\mathrm{f}(m, param) = 0$ then	$\quad$ If $C.\mathrm{f}(m, param) = 1$ then
$\quad\quad$ Return m	$\quad\quad$ Return m
Return m	Return m

Figure 69: **$IND-CPA-KI adversary D and constructed sampler $\mathcal{M}_0, \mathcal{M}_1$ in the proof of Theorem C.1.**

Thus, $\mathbf{Adv}^{\$\text{sim-cpa-ki}}_{\mathsf{PKE},B,S_B,\mathcal{M}}(k) \leq \Big(2/\delta + 1\Big)^2 \cdot \mathbf{Adv}^{\$\text{sim-cpa-ki}}_{\mathsf{PKE},C,S_C,\mathcal{M}}(k)$. This completes the proof of Claim C.3.

Finally, we conclude the proof with following claim.

Claim C.4 Let $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a R-PKE scheme. Let C be a δ-*balanced* boolean $SIM-CPA-KI adversary against PKE with respect to the message sampler $\mathcal{M}$. Then there is a $IND-CPA-KI adversary D such that for all $k \in \mathbb{N}$

$$\mathbf{Adv}^{\$\text{sim-cpa-ki}}_{\mathsf{PKE},C,S,\mathcal{M}}(k) \leq \frac{1}{2}\mathbf{Adv}^{\$\text{ind-cpa-ki}}_{\mathsf{PKE},D,\mathcal{M}_0,\mathcal{M}_1}(k) + \frac{1}{2}\Big(\frac{1}{2} + \delta\Big)^k \ .$$

where $\mathcal{M}_0, \mathcal{M}_1$ are $\log(1/(1/2 - \delta))$-induced distributions of $\mathcal{M}$ and the running time of D is about that k-times of C.

Proof of Claim C.4. We define $E^{R,1}_D$ and $E^{L,1}_D$ to be the events that game $IND-CPA-KI-LEFT$^{D,\mathcal{M}_0,\mathcal{M}_1}_{\mathsf{PKE}}$ and $IND-CPA-KI-RIGHT$^{D,\mathcal{M}_0,\mathcal{M}_1}_{\mathsf{PKE}}$ output 1, respectively. We also define $E^{R,0}_D$ and $E^{L,0}_D$ to be the events that game $IND-CPA-KI-LEFT$^{D,\mathcal{M}_0,\mathcal{M}_1}_{\mathsf{PKE}}$ and $IND-CPA-KI-RIGHT$^{D,\mathcal{M}_0,\mathcal{M}_1}_{\mathsf{PKE}}$ outputs 0, respectively. Consider adversary D attacking PKE and message samplers $\mathcal{M}_0, \mathcal{M}_1$ shown in Figure 69. Therefore, we have $\mathbf{Adv}^{\$\text{ind-cpa-ki}}_{\mathsf{PKE},D,\mathcal{M}_0,\mathcal{M}_1}(k) = \Pr\Big[E^{R,1}_D\Big] - \Pr\Big[E^{L,1}_D\Big]$. Let T to be the event that the final return statements of $\mathcal{M}_0$ and $\mathcal{M}_1$ are executed and let $X = \Pr\Big[E^{R,1}_D \mid \overline{T}\Big] + \Pr\Big[E^{L,0}_D \mid \overline{T}\Big]$. Hence,

$$\begin{aligned}
\mathbf{Adv}^{\$\text{ind-cpa-ki}}_{\mathsf{PKE},D,\mathcal{M}_0,\mathcal{M}_1}(k) &\geq \Big(\Pr\Big[E^{R,1}_D \mid \overline{T}\Big] - \Pr\Big[E^{L,1}_D \mid \overline{T}\Big]\Big) \cdot \Pr\big[\overline{T}\big] && (1)\\
&= \Big(\Pr\Big[E^{R,1}_D \mid \overline{T}\Big] + \Pr\Big[E^{L,0}_D \mid \overline{T}\Big] - 1\Big) \cdot \Pr\big[\overline{T}\big] && (2)\\
&= \Big(X - 1\Big) \cdot \Pr\big[\overline{T}\big] \ . && (3)
\end{aligned}$$

We define E^1_C and E^0_C to be the events that game $SIM-CCA-KI-REAL$^{C,\mathcal{M}}_{\mathsf{PKE}}$ and $SIM-CCA-KI-IDEAL$^{C,S,\mathcal{M}}_{\mathsf{PKE}}$ outputs 1, respectively. Consider the simulator S_C as shown in Figure 70. Note that,

$$\Pr\big[E^1_C\big] = \Pr\big[C.\mathrm{f}(m) = 1\big] \cdot \Pr\Big[E^{R,1}_D \mid \overline{T}\Big] + \Pr\big[C.\mathrm{f}(m) = 0\big] \cdot \Pr\Big[E^{L,0}_D \mid \overline{T}\Big] \ .$$

Assume $\Pr\big[C.\mathrm{f}(m) = 1\big] = 1/2 - \epsilon$, for some $\epsilon \in [-1/2, 1/2]$ and let $Y = \Pr\Big[E^{R,1}_D \mid \overline{T}\Big] - \Pr\Big[E^{L,0}_D \mid \overline{T}\Big]$. Therefore, we have $\Pr\big[E^1_C\big] = X/2 + \epsilon \cdot Y$. On the other hand, we have $\Pr\big[E^0_C\big] = \sum_b \Pr\big[C.\mathrm{f}(m') = b\big] \cdot \Pr\big[C.\mathrm{g}(pk, c) = b\big]$.

Simulator $S_C(pk, param, \|m\|)$
$m' \leftarrow\!\!\$\ \mathcal{M}(1^k, param)$
$c' \leftarrow \mathsf{Enc}(pk, m')$
$\omega \leftarrow\!\!\$\ A.\mathrm{g}(pk, c', param)$
Return ω

Figure 70: **Constructed simulator S_C in the proof of Theorem C.1.**

We also have,

$$\begin{aligned}\Pr[C.\mathrm{g}(pk,c)=0] &= \sum_{b=0}^{1}\Pr[C.\mathrm{f}(m)=b]\cdot\Pr[C.\mathrm{g}(pk,c)=0\mid C.\mathrm{f}(m)=b]\\ &= \Pr[C.\mathrm{f}(m)=0]\cdot\Pr\left[E_D^{L,0}\mid\overline{T}\right]+\Pr[C.\mathrm{f}(m)=1]\cdot\left(1-\Pr\left[E_D^{R,1}\mid\overline{T}\right]\right)\\ &= \frac{1}{2}(Y+1)+(X-1)\cdot\epsilon\ .\end{aligned}$$

Similarly, we have $\Pr[C.\mathrm{g}(pk,c)=1] = 1/2(1-Y)+(1-X)\cdot\epsilon$. Thus,

$$\mathbf{Adv}^{\text{\$sim-cpa-ki}}_{\mathsf{PKE},C,S,\mathcal{M}}(k) = (\frac{1}{2}-2\epsilon^2)(X-1) \tag{4}$$

$$\leq \frac{1}{2}(X-1)\ . \tag{5}$$

From equations 3 and 5, we obtain that $\mathbf{Adv}^{\text{\$ind-cpa-ki}}_{\mathsf{PKE},D,\mathcal{M}_0,\mathcal{M}_1}(k) \geq 2\mathbf{Adv}^{\text{\$sim-cpa-ki}}_{\mathsf{PKE},C,S,\mathcal{M}}(k)\cdot\Pr[\overline{T}]$. Moreover, we know that C is a δ-*balanced* adversary. Then,

$$\begin{aligned}\mathbf{Adv}^{\text{\$ind-cpa-ki}}_{\mathsf{PKE},D,\mathcal{M}_0,\mathcal{M}_1}(k) &\geq 2\mathbf{Adv}^{\text{\$sim-cpa-ki}}_{\mathsf{PKE},C,S,\mathcal{M}}(k)\Big(1-\big(\frac{1}{2}+\delta\big)^k\Big)\\ &\geq 2\mathbf{Adv}^{\text{\$sim-cpa-ki}}_{\mathsf{PKE},C,S,\mathcal{M}}(k)-\Big(\frac{1}{2}+\delta\Big)^k\ .\end{aligned}$$

Setting $\delta = 1/4$, will conclude the proof of Theorem C.1. ∎

D Security of s-clear RSA-OAEP in the RO Model

We show that s-clear RSA-OAEP is IND-CCA2 secure in random oracle model. As a warm-up (which is useful in the final result), we begin by showing that s-clear RSA-OAEP is IND-CPA secure.

Theorem D.1 Let μ, ζ, ρ be integer parameters. Let $\mathcal{F}$ be a family of one-way trapdoor permutations with domain $\{0,1\}^\rho$. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ be a RO and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ be a function family. Then $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}[\mathcal{G},\mathcal{H},\mathcal{F}]$ is IND-CPA secure in the random oracle model. In particular, for any adversary A, there is an adversary B such that

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}},A}(k) \leq \frac{2q}{2^\rho} + 2\cdot\mathbf{Adv}^{\text{owf}}_{\mathcal{F},B}(k)\ .$$

where q is the total number of random-oracle queries of A and $\mathcal{M}$. Adversary B makes at most q random-oracle queries. The running time of B is about that of A.

Proof: Consider games G_1–G_4 in Figure 71. Each game maintains two independent random oracles RO and $\overline{\mathrm{RO}}$. Procedure RO maintains a local array G as follows:

Procedure $\mathrm{RO}(v)$
If $G[v] = \perp$ then $G[v] \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$
Return $G[v]$

Games $G_1(k)$, $G_2(k)$	**Games** $G_3(k)$, $G_4(k)$
$b \leftarrow\!\!\$\ \{0,1\}$; $K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$	$b \leftarrow\!\!\$\ \{0,1\}$; $K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_H, f)$	$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_H, f)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{RO}_1(\cdot)}(1^k, pk)$	$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{RO}_1(\cdot)}(1^k, pk)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{RO}_1(\cdot)}(1^k, pk)$; $r \leftarrow\!\!\$\ \{0,1\}^\rho$	$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{RO}_1(\cdot)}(1^k, pk)$; $r \leftarrow\!\!\$\ \{0,1\}^\rho$
If $G[r] \neq \bot$ then	$s \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$; $x \leftarrow s \oplus (m_b \| 0^\zeta)$
$\quad \mathsf{bad}_1 \leftarrow \mathsf{true}$; $G[r] \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$	$G[r] \leftarrow x$; $z \leftarrow H(K_H, s)$; $t \leftarrow z \oplus r$
Else $G[r] \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$	$y \leftarrow f(t)$; $c \leftarrow (s, y)$
$x \leftarrow G[r]$; $s \leftarrow x \oplus (m_b \| 0^\zeta)$	$d \leftarrow\!\!\$\ A_2^{\mathrm{RO}_2(\cdot)}(c, state)$
$z \leftarrow H(K_H, s)$; $t \leftarrow z \oplus r$	Return $(b = d)$
$y \leftarrow f(t)$; $c \leftarrow (s, y)$	**Procedure** $\mathrm{RO}_1(v)$
$d \leftarrow\!\!\$\ A_2^{\mathrm{RO}_2(\cdot)}(c, state)$	Return $\mathrm{RO}(v)$
Return $(b = d)$	**Procedure** $\mathrm{RO}_2(v)$
Procedure $\mathrm{RO}_1(v)$	If $v = r$ then
Return $\mathrm{RO}(v)$	$\quad \mathsf{bad}_2 \leftarrow \mathsf{true}$; return $\overline{\mathrm{RO}}(v)$
Procedure $\mathrm{RO}_2(v)$	Return $\mathrm{RO}(v)$
Return $\mathrm{RO}(v)$	

Figure 71: **Games G_1–G_4 in the proof of Theorem D.1.**

Algorithm $B(f, y)$	**Procedure** $\mathrm{ROSIM}_1(v)$
$K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$; $\mathsf{out} \leftarrow \bot$	If $G[v] = \bot$ then $G[v] \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$
$pk \leftarrow (K_H, f)$; $b \leftarrow\!\!\$\ \{0,1\}$	Return $G[v]$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\mathrm{ROSIM}_1(\cdot)}(1^k, pk)$	**Procedure** $\mathrm{ROSIM}_2(v)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{ROSIM}_1(\cdot)}(1^k, pk)$	If $f(v \oplus z) = y$ then
$s \leftarrow \{0,1\}^{\mu+\zeta}$; $z \leftarrow H(K_H, s)$	$\quad \mathsf{out} \leftarrow v$; Halt run of A_2
$c \leftarrow (s, y)$	If $G[v] = \bot$ then $G[v] \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$
Run $A_2^{\mathrm{ROSIM}_2(\cdot)}(c, state)$	Return $G[v]$
Return out	

Figure 72: **Adversary B in the proof of Theorem D.1.**

For simplicity, we omit the code of $\mathrm{RO}, \overline{\mathrm{RO}}$ in the games. In each game, we use RO_1 to denote the oracle interface of adversary A_1 and message samplers $\mathcal{M}_0, \mathcal{M}_1$ and we use RO_2 to denote the oracle interface of adversary A_2. Game G_1 corresponds to game $\text{IND-CPA}^A_{\mathsf{OAEP}_{\mathsf{s-clear}}}$. Then

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}_{\mathsf{s-clear}}, A}(k) = 2 \cdot \Pr[G_1(k) \Rightarrow 1] - 1 \ .$$

We now explain the game chain. Game G_2 is identical to game G_1, except in the encryption of message m_b, if either adversary A_1 or message sampler $\mathcal{M}_b$ queried r to their random oracle RO_1, then it chooses a fresh random value for $G[r]$. Games G_1 and G_2 are identical-until-bad_1, and thus from the Fundamental Lemma of Game-playing [14],

$$\Pr\left[\, G_1(k) \Rightarrow 1 \,\right] - \Pr\left[\, G_2(k) \Rightarrow 1 \,\right] \leq \Pr\left[\, G_2(k) \text{ sets } \mathsf{bad}_1 \,\right] \leq \frac{q}{2^\rho} \ .$$

In game G_3, we reorder the code of game G_2 in producing s. The change is conservative, meaning that $\Pr[G_2(k) \Rightarrow 1] = \Pr[G_3(k) \Rightarrow 1]$. Game G_4 is identical to game G_3, except in procedure RO_2, if adversary A_2 make a query for r, then the oracle lies, calling $\overline{\mathrm{RO}}$ instead. Game G_3 and game G_4 are identical-until-bad_2, and based on Fundamental Lemma of Game-playing [14],

$$\Pr\left[\, G_3(k) \Rightarrow 1 \,\right] - \Pr\left[\, G_4(k) \Rightarrow 1 \,\right] \leq \Pr\left[\, G_4(k) \text{ sets } \mathsf{bad}_2 \,\right] \ .$$

Consider adversary B attacking trapdoor permutation $\mathcal{F}$ in Figure 72. Then,

Games $G_1(k)$, $G_2(k)$	**Procedure** $\textsc{Dec}(c')$ // of game G_1
$b \leftarrow\!\!\$\ \{0,1\}$; $\mathsf{Dom} \leftarrow \emptyset$; $K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$	$sk \leftarrow (K_H, f^{-1})$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_H, f)$	$m' \leftarrow \mathsf{Dec}(sk, c')$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\textsc{Dec}(\cdot),\textsc{ROSim}(\cdot)}(1^k, pk)$	Return m'
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\mathrm{RO}(\cdot)}(1^k, pk)$; $r \leftarrow\!\!\$\ \{0,1\}^\rho$	**Procedure** $\textsc{Dec}(c')$ // of game G_2
$x \leftarrow \mathrm{RO}(r)$; $s \leftarrow x \oplus (m_b \| 0^\zeta)$	$(s', y') \leftarrow c'$
$z \leftarrow H(K_H, s)$; $t \leftarrow z \oplus r$	For $r \in \mathsf{Dom}$ do
$y \leftarrow f(t)$; $c \leftarrow (s, y)$	If $\mathrm{RO}(r)\|_\zeta = s'\|_\zeta$ then
$d \leftarrow\!\!\$\ A_2^{\textsc{Dec}(\cdot),\textsc{ROSim}(\cdot)}(c, state)$	$m^* \leftarrow s' \oplus \mathrm{RO}(r)$; $m' \leftarrow m^*\|^\mu$
Return $(b = d)$	If $c' = \mathsf{Enc}(pk, m'; r)$ then
Procedure $\textsc{ROSim}(r)$	Return m'
$\mathsf{Dom} \leftarrow \mathsf{Dom} \cup \{r\}$	Return $\perp$
Return $\mathrm{RO}(r)$	

Figure 73: **Games G_1 and G_2 of the proof of Theorem D.2.**

$$\Pr[\, G_4(k) \text{ sets } \mathsf{bad}_2 \,] \le \mathbf{Adv}^{\mathrm{owf}}_{\mathcal{F},B}(k) \ .$$

Note that $\Pr[G_4(k) \Rightarrow 1] = 1/2$, since the distribution of the ciphertexts are completely independent of the bit b. Summing up,

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}},A}(k) \le \frac{2q}{2^\rho} + 2 \cdot \mathbf{Adv}^{\mathrm{owf}}_{\mathcal{F},B}(k)$$

This completes the proof. ▌

To achieve IND-CCA2 security, we need additional assumptions on $\mathcal{H}$ and $\mathcal{F}$. In particular, we need $\mathcal{H}$ to be collision-resistant and trapdoor permutation family $\mathcal{F}$ to be XOR-non-malleable. (The latter is also necessary in general due to [70].)

Theorem D.2 Let μ, ζ, ρ be integer parameters. Let $\mathcal{F}$ be a family of trapdoor permutations with domain $\{0,1\}^\rho$. Suppose $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ is a RO and $\mathcal{H} : \mathcal{K}_H \times \{0,1\}^{\mu+\zeta} \to \{0,1\}^\rho$ is collision-resistant. Suppose $\mathcal{F}$ is XOR-NM. Then $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ is IND-CCA2 secure in the random oracle model. In particular, for any adversary A, there are adversaries B, C and D such that

$$\mathbf{Adv}^{\text{ind-cca2}}_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}},A}(k) \le \frac{2q}{2^\rho} + \frac{4p}{2^\zeta} + 2 \cdot \mathbf{Adv}^{\mathrm{owf}}_{\mathcal{F},B}(k) + 2 \cdot \mathbf{Adv}^{\mathrm{cr}}_{\mathcal{H},C}(k) + 2 \cdot \mathbf{Adv}^{\text{xor-nm0}}_{\mathcal{F},D}(k) \ .$$

where p is the number of decryption-oracle queries of A and q is the total number of random-oracle queries of A and $\mathcal{M}$. Adversary B, C and D makes at most q random-oracle queries. The running time of B, C and D are about that of A.

Proof: Consider games G_1,G_2 in Figures 73. Then

$$\mathbf{Adv}^{\text{ind-cca2}}_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}},A}(k) = 2 \cdot \Pr[G_1(k) \Rightarrow 1] - 1 \ .$$

Game G_2 is identical to game G_1, except for the following. In procedure $\textsc{Dec}(c)$, instead of using the decryption of $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ to decrypt c, we maintain the set Dom of random-oracle queries r that adversaries A_1 and A_2 make. If there is $r \in \mathsf{Dom}$ and a message m such that c is the corresponding ciphertext of m under randomness r, then we return m; otherwise return $\perp$. Wlog, assume that A_1 stores all random-oracle queries/answers in its state; that is, both A_1 and A_2 also can track Dom and implement the $\textsc{Dec}$ procedure of game G_2 on their own, without calling the decryption oracle. The adversaries can distinguish the games if and only if they can trigger $\textsc{Dec}$ of game G_1 to produce non-$\perp$ output.

For $i \in [p]$, let $c_i = (s_i, y_i)$ be the i-th decryption queries made by A and $r_i = H(K_H, s_i) \oplus f^{-1}(y_i)$. Note that in the game G_2 when $r_i \in \mathsf{Dom}$, adversary A always get the correct plaintext. Therefore, adversary A can distinguish the games if and only if there exist c_i such that c_i is a valid ciphertext and $r_i \notin \mathsf{Dom}$. Let $c = (s, y)$ be the challenge ciphertext and r be the corresponding randomness. We define T to be the event that there exists at least one decryption query c_i such that the corresponding randomness $r_i \notin \mathsf{Dom} \cup \{r\}$ and $\mathrm{RO}(r_i)|_\zeta = s_i|_\zeta$. We also define R to be the event that there exists at least one decryption query c_i such that the corresponding

Algorithm $C(K_H)$

$\mathsf{out}_1 \leftarrow \bot$; $\mathsf{out}_2 \leftarrow \bot$; $b \leftarrow\!\!\$\ \{0,1\}$
$(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$; $pk \leftarrow (K_H, f)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\text{DEC}(\cdot),\text{ROSIM}(\cdot)}(1^k, pk)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\text{RO}_1(\cdot)}(1^k, pk)$; $r \leftarrow\!\!\$\ \{0,1\}^\rho$
$x \leftarrow \text{RO}(r)$; $s \leftarrow x \oplus (m_b || 0^\zeta)$
$z \leftarrow H(K_H, s)$; $t \leftarrow z \oplus r$
$\mathsf{out}_1 \leftarrow s$; $y \leftarrow f(t)$; $c \leftarrow (s, y)$
Run $A_2^{\text{DEC}(\cdot),\text{ROSIM}(\cdot)}(c, state)$
Return $\mathsf{out}_1, \mathsf{out}_2$

Procedure ROSIM(r)

$\mathsf{Dom} \leftarrow \mathsf{Dom} \cup \{r\}$
Return $\text{RO}(r)$

Procedure DEC(c')

$(s', y') \leftarrow c'$
$r' \leftarrow f^{-1}(y') \oplus H(K_H, s')$
If $s \neq s'$ and $s|_\zeta = s'|_\zeta$ and $y = y'$ and $r = r'$
  $\mathsf{out}_2 \leftarrow s'$; Halt A
For $r \in \mathsf{Dom}$ do
  If $\text{RO}(r)|_\zeta = s'|_\zeta$ then
    $m^* \leftarrow s' \oplus \text{RO}(r)$; $m' \leftarrow m^*|^\mu$
    If $m^*|_\zeta = 0^\zeta$ and $c' = \mathsf{Enc}(pk, m'; r)$ then
      Return m'
Return $\bot$

Figure 74: **Adversary C in the proof of Theorem D.2.**

Algorithm $D(f, y)$

$\mathsf{out}_1 \leftarrow \bot$; $\mathsf{out}_2 \leftarrow \bot$; $b \leftarrow\!\!\$\ \{0,1\}$
$K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$; $pk \leftarrow (K_H, f)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\ A_1^{\text{DEC}(\cdot),\text{ROSIM}(\cdot)}(1^k, pk)$
$m_b \leftarrow\!\!\$\ \mathcal{M}_b^{\text{RO}_1(\cdot)}(1^k, pk)$; $x \leftarrow\!\!\$\ \{0,1\}^{\mu+\zeta}$
$s \leftarrow x \oplus (m_b || 0^\zeta)$; $c \leftarrow (s, y)$
Run $A_2^{\text{DEC}(\cdot),\text{ROSIM}(\cdot)}(c, state)$
Return $(\mathsf{out}_1, \mathsf{out}_2)$

Procedure ROSIM(r)

$\mathsf{Dom} \leftarrow \mathsf{Dom} \cup \{r\}$
Return $\text{RO}(r)$

Procedure DEC(c')

$(s', y') \leftarrow c'$
If $s \neq s'$ and $s|_\zeta = s'|_\zeta$ and $y \neq y'$
  $\mathsf{out}_1 \leftarrow H(K_H, s) \oplus H(K_H, s')$
  $\mathsf{out}_2 \leftarrow y'$; Halt A
For $r \in \mathsf{Dom}$ do
  If $\text{RO}(r)|_\zeta = s'|_\zeta$ then
    $m^* \leftarrow s' \oplus \text{RO}(r)$; $m' \leftarrow m^*|^\mu$
    If $m^*|_\zeta = 0^\zeta$ and $c' = \mathsf{Enc}(pk, m'; r)$
      Return m'
Return $\bot$

Figure 75: **Adversary D in the proof of Theorem D.2.**

randomness $r_i = r$ and $s|_\zeta = s_i|_\zeta$. Note that when T or R happen, adversary A can distinguish game G_1 and game G_2. Then

$$\Pr[G_1(k) \Rightarrow 1] - \Pr[G_2(k) \Rightarrow 1] \leq \Pr[T] + \Pr\left[\overline{T} \wedge R\right] .$$

Moreover, we know for any decryption query c_i, when there is no prior random-oracle query on r_i, $\text{RO}(r_i)|_\zeta$ is equal to $s_i|_\zeta$ with probability $2^{-\zeta}$. Multiplying for p decryption-oracle queries, we obtain $\Pr[T] \leq p/2^\zeta$. Next, we define E_1 to be the event such that there exists at least one decryption query c_i, where $r_i = r$, $s \neq s_i$, $s|_\zeta = s_i|_\zeta$ and $y = y_i$. We also define E_2 to be the event such that there exists at least one decryption query c_i where $r_i = r$, $s \neq s_i$, $s|_\zeta = s_i|_\zeta$ and $y \neq y_i$ and E_3 to be the event such that there exists at least one decryption query c_i where the corresponding $r_i = r$, $s = s_i$ and $y \neq y_i$. Then, we obtain $\Pr\left[\overline{T} \wedge R\right] \leq \Pr\left[\overline{T} \wedge E_1\right] + \Pr\left[\overline{T} \wedge E_2\right] + \Pr\left[\overline{T} \wedge E_3\right]$.

Note that when E_1 happens, there exists at least one decryption query c_i such that $r_i = r$, $s \neq s_i$, $s|_\zeta = s_i|_\zeta$ and $y = y_i$. Consider adversary C attacking the hash function family $\mathcal{H}$ in Figure 74. Note that when adversary A_2 makes such decryption query, adversary C is able to find a collision. Then, we obtain $\Pr[E_1] \leq \mathbf{Adv}^{\text{cr}}_{\mathcal{H},C}(k)$.

Next, when E_2 happen, there exists at least one decryption query c_i such that $r_i = r$, $s \neq s_i$, $s|_\zeta = s_i|_\zeta$ and $y \neq y_i$. We define S to be the event such that there exists at least one decryption query c_i where the corresponding $r_i \neq r$ and $s|_\zeta = \text{RO}(r_i)|_\zeta$. Thus, we obtain $\Pr\left[\overline{T} \wedge E_2\right] = \Pr\left[\overline{T} \wedge E_2 \wedge S\right] + \Pr\left[\overline{T} \wedge E_2 \wedge \overline{S}\right]$. Note that when event S happens, we get a collision on random oracle $\mathcal{G}$. Multiplying for p decryption-oracle queries, we get $\Pr\left[\overline{T} \wedge E_2 \wedge S\right] \leq p/2^\zeta$. On the other hand, consider XOR-NM0 adversary D attacking the trapdoor permutation family $\mathcal{F}$ in Figure 75. Note that when $\overline{T}, E_2$ and $\overline{S}$ happen, XOR-NM0 adversary D is able to successfully attack $\mathcal{F}$. Then, we obtain $\Pr\left[\overline{T} \wedge E_2 \wedge S\right] \leq \mathbf{Adv}^{\text{xor-nm0}}_{\mathcal{F},D}(k)$.

Moreover, when E_3 happens, there exists at least one decryption query c_i such that $r_i = r$, $s = s_i$ and $y \neq y_i$. Note that this is impossible to happen since when $s = s_i$ and $y \neq y_i$, the corresponding randomness $r_i \neq r$. Now

in game G_2, the decryption oracle always return correct answer whp. Thus wlog, assume that adversary A never makes any decryption query, meaning that they only launch a IND-CPA attack. Hence

$$\Pr[G_2(k) \Rightarrow 1] \leq \frac{1}{2} + \frac{1}{2}\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}},A}(k) \ .$$

From Theorem D.1

$$\mathbf{Adv}^{\text{ind-cca2}}_{\mathsf{OAEP}_{\mathsf{s\text{-}clear}},A}(k) \leq \frac{2q}{2^\rho} + \frac{4p}{2^\zeta} + 2 \cdot \mathbf{Adv}^{\text{owf}}_{\mathcal{F},B}(k) + 2 \cdot \mathbf{Adv}^{\text{cr}}_{\mathcal{H},C}(k) + 2 \cdot \mathbf{Adv}^{\text{xor-nm0}}_{\mathcal{F},D}(k) \ .$$

This completes the proof. ∎

E Deferred Proofs

E.1 Proof of Lemma 7.11

Consider the games G_1–G_7 in Figure 76. Note that game G_1 is identical to EXT2 game with hint function $\mathsf{OAEP}_{\mathsf{s\text{-}clear}}[\mathcal{G}, \mathcal{H}, \mathcal{F}|^{\mu+\zeta}]$. Then,

$$\Pr[G_1(k) \Rightarrow 1] = \mathbf{Adv}^{\eta\text{-ext2}_\zeta}_{\mathcal{G},\mathsf{OAEP}_{\mathsf{s\text{-}clear}},A,\mathcal{E}}(k) \ .$$

We now explain the game chain. Game G_2 is identical to game G_1, except in the image oracle, instead of using value $H(K_H, s) \oplus r$ as t, we use value r. Consider adversary C_ℓ for $\ell \in [p]$ in Figure 77. We have $\Pr[G_1(k) \Rightarrow 1] - \Pr[G_2(k) \Rightarrow 1] \leq \sum_\ell \mathbf{Adv}^{\text{xor-ind2}_\zeta}_{\mathcal{F},\mathcal{G},C_\ell}(k)$. Wlog, we can assume there exist adversary C such that for all $\ell \in [p]$, we have $\mathbf{Adv}^{\text{xor-ind2}_\zeta}_{\mathcal{F},\mathcal{G},C_\ell}(k) \leq \mathbf{Adv}^{\text{xor-ind2}_\zeta}_{\mathcal{F},\mathcal{G},C}(k)$. Then we obtain

$$\Pr[G_1(k) \Rightarrow 1] - \Pr[G_2(k) \Rightarrow 1] \leq p \cdot \mathbf{Adv}^{\text{xor-ind2}_\zeta}_{\mathcal{F},\mathcal{G},C}(k) \ .$$

Next, game G_3 is identical to game G_2, except in the image oracle, we use uniformly random value in producing t instead of value r. Note that this is very similar to switching game G_1 to G_2. For simplicity, we omit adversary's details here. Then,

$$\Pr[G_2(k) \Rightarrow 1] - \Pr[G_3(k) \Rightarrow 1] \leq p \cdot \mathbf{Adv}^{\text{xor-ind2}_\zeta}_{\mathcal{F},\mathcal{G},C}(k) \ .$$

Next, game G_4 is identical to game G_3, except in the image oracle, we use uniformly random value as x instead of value $G(K_G, r)$. Consider adversary D_ℓ for $\ell \in [p]$ in Figure 78. We have $\Pr[G_3(k) \Rightarrow 1] - \Pr[G_4(k) \Rightarrow 1] \leq \sum_\ell \mathbf{Adv}^{\text{vprg}_\zeta}_{\mathcal{G},D_\ell}(k)$. Wlog, we can assume there exist adversary D such that for all $\ell \in [p]$, we have $\mathbf{Adv}^{\text{vprg}_\zeta}_{\mathcal{G},D_\ell}(k) \leq \mathbf{Adv}^{\text{vprg}_\zeta}_{\mathcal{G},D}(k)$. Then we obtain

$$\Pr[G_3(k) \Rightarrow 1] - \Pr[G_4(k) \Rightarrow 1] \leq p \cdot \mathbf{Adv}^{\text{vprg}_\zeta}_{\mathcal{G},D}(k) \ .$$

In game G_5, we reorder the code of game G_4 in producing s. The change is conservative, meaning that $\Pr[G_4(k) \Rightarrow 1] = \Pr[G_5(k) \Rightarrow 1]$. Game G_6 is identical to game G_5, except in the image oracle, we use value $G(K_G, r)$ as x instead of uniformly random value. Note that this is very similar to switching game G_3 to G_4. For simplicity, we omit adversary's details here. Then,

$$\Pr[G_5(k) \Rightarrow 1] - \Pr[G_6(k) \Rightarrow 1] \leq p \cdot \mathbf{Adv}^{\text{vprg}_\zeta}_{\mathcal{G},D}(k) \ .$$

Next, game G_7 is identical to game G_6, except in the image oracle, we use value r in producing t instead of uniformly random value. Note that this is very similar to switching game G_1 to G_2. For simplicity, we omit adversary's details here. Then,

$$\Pr[G_6(k) \Rightarrow 1] - \Pr[G_7(k) \Rightarrow 1] \leq p \cdot \mathbf{Adv}^{\text{xor-ind2}_\zeta}_{\mathcal{F},\mathcal{G},C}(k) \ .$$

Let w be the randomness of adversary A in the game EXT2. Let K_H be the key for the hash function family $\mathcal{H}$ in the game EXT2. We define EXT2 adversary B with the hint function $\mathcal{F}$ and randomness w in Figure 79. Let K_H be the key independent auxiliary input to adversary B. Note that auxiliary input K_H is independent of key K_G. Note that, $\Pr[G_7(k) \Rightarrow 1] = \mathbf{Adv}^{\delta\text{-ext2}_\zeta}_{\mathcal{G},\mathcal{F},B,\mathcal{E}}(k)$.

Games $G_1(k)$, $G_2(k)$	**Games** $G_3(k)$, $G_4(k)$
$i \leftarrow 1$; $j \leftarrow 1$; $w \leftarrow\$ \mathsf{Coins}(k)$; $\mathsf{out} \leftarrow 0$	$i \leftarrow 1$; $j \leftarrow 1$; $w \leftarrow\$ \mathsf{Coins}(k)$; $\mathsf{out} \leftarrow 0$
$\mathbf{r} \leftarrow \varepsilon$; $\mathbf{s}_2 \leftarrow \varepsilon$; $\mathbf{x} \leftarrow \varepsilon$; $\mathbf{c} \leftarrow \varepsilon$	$\mathbf{r} \leftarrow \varepsilon$; $\mathbf{s}_2 \leftarrow \varepsilon$; $\mathbf{x} \leftarrow \varepsilon$; $\mathbf{c} \leftarrow \varepsilon$
$K_G \leftarrow\$ \mathcal{K}_G(1^k)$; $K_H \leftarrow\$ \mathcal{K}_H(1^k)$	$K_G \leftarrow\$ \mathcal{K}_G(1^k)$; $K_H \leftarrow\$ \mathcal{K}_H(1^k)$
$(f, f^{-1}) \leftarrow\$ \mathsf{Kg}(1^k)$; $state \leftarrow (K_G, K_H, f)$	$(f, f^{-1}) \leftarrow\$ \mathsf{Kg}(1^k)$; $state \leftarrow (K_G, K_H, f)$
Run $A^{\mathcal{O}(\cdot),\mathcal{I}(\cdot)}(K_G, K_H, f; w)$	Run $A^{\mathcal{O}(\cdot),\mathcal{I}(\cdot)}(K_G, K_H, f; w)$
For $i = 1$ to $\lvert\mathbf{r}\rvert$ do	For $i = 1$ to $\lvert\mathbf{r}\rvert$ do
If $\exists v\colon G(K_G, v)\rvert_\zeta = \mathbf{s}_2[i]$	If $\exists v\colon G(K_G, v)\rvert_\zeta = \mathbf{s}_2[i]$
If $G(K_G, \mathbf{r}[i])\rvert_\zeta \neq \mathbf{s}_2[i]$ then $\mathsf{out} \leftarrow 1$	If $G(K_G, \mathbf{r}[i])\rvert_\zeta \neq \mathbf{s}_2[i]$ then $\mathsf{out} \leftarrow 1$
Return out	Return out
Procedure $\mathcal{O}(x)$	**Procedure** $\mathcal{O}(x)$
If $x \in \mathbf{x}$ then return $\perp$	If $x \in \mathbf{x}$ then return $\perp$
$(state, r) \leftarrow \mathcal{E}(state, \mathbf{x}, \mathbf{c}, x; w)$	$(state, r) \leftarrow \mathcal{E}(state, \mathbf{x}, \mathbf{c}, x; w)$
$\mathbf{r}[i] \leftarrow r$; $\mathbf{s}_2[i] \leftarrow x$; $i \leftarrow i + 1$	$\mathbf{r}[i] \leftarrow r$; $\mathbf{s}_2[i] \leftarrow x$; $i \leftarrow i + 1$
Return r	Return r
Procedure $\mathcal{I}(\mathcal{M})$	**Procedure** $\mathcal{I}(\mathcal{M})$
$m \leftarrow\$ \mathcal{M}(1^k)$; $r \leftarrow\$ \mathsf{GDom}(k)$	$m \leftarrow\$ \mathcal{M}(1^k)$; $r \leftarrow\$ \mathsf{GDom}(k)$
$x \leftarrow G(K_G, r)$; $s \leftarrow x \oplus m \Vert 0^\zeta$	$x \leftarrow G(K_G, r)$; $x \leftarrow\$ \mathsf{GRng}(k)$; $s \leftarrow x \oplus m \Vert 0^\zeta$
$t \leftarrow H(K_H, s) \oplus r$; $t \leftarrow r$; $y \leftarrow f(t)$	$t \leftarrow\$ \mathsf{GDom}(k)$; $y \leftarrow f(t)$
$\mathbf{x}[j] \leftarrow x\rvert_\zeta$; $\mathbf{c}[j] \leftarrow (s, y)$; $j \leftarrow j + 1$	$\mathbf{x}[j] \leftarrow x\rvert_\zeta$; $\mathbf{c}[j] \leftarrow (s, y)$; $j \leftarrow j + 1$
Return (s, c)	Return (s, c)

Games $G_5(k)$, $G_6(k)$	**Games** $G_7(k)$
$i \leftarrow 1$; $j \leftarrow 1$; $w \leftarrow\$ \mathsf{Coins}(k)$; $\mathsf{out} \leftarrow 0$	$i \leftarrow 1$; $j \leftarrow 1$; $w \leftarrow\$ \mathsf{Coins}(k)$; $\mathsf{out} \leftarrow 0$
$\mathbf{r} \leftarrow \varepsilon$; $\mathbf{s}_2 \leftarrow \varepsilon$; $\mathbf{x} \leftarrow \varepsilon$; $\mathbf{c} \leftarrow \varepsilon$	$\mathbf{r} \leftarrow \varepsilon$; $\mathbf{s}_2 \leftarrow \varepsilon$; $\mathbf{x} \leftarrow \varepsilon$; $\mathbf{c} \leftarrow \varepsilon$
$K_G \leftarrow\$ \mathcal{K}_G(1^k)$; $K_H \leftarrow\$ \mathcal{K}_H(1^k)$	$K_G \leftarrow\$ \mathcal{K}_G(1^k)$; $K_H \leftarrow\$ \mathcal{K}_H(1^k)$
$(f, f^{-1}) \leftarrow\$ \mathsf{Kg}(1^k)$; $state \leftarrow (K_G, K_H, f)$	$(f, f^{-1}) \leftarrow\$ \mathsf{Kg}(1^k)$; $state \leftarrow (K_G, K_H, f)$
Run $A^{\mathcal{O}(\cdot),\mathcal{I}(\cdot)}(K_G, K_H, f; w)$	Run $A^{\mathcal{O}(\cdot),\mathcal{I}(\cdot)}(K_G, K_H, f; w)$
For $i = 1$ to $\lvert\mathbf{r}\rvert$ do	For $i = 1$ to $\lvert\mathbf{r}\rvert$ do
If $\exists v\colon G(K_G, v)\rvert_\zeta = \mathbf{s}_2[i]$	If $\exists v\colon G(K_G, v)\rvert_\zeta = \mathbf{s}_2[i]$
If $G(K_G, \mathbf{r}[i])\rvert_\zeta \neq \mathbf{s}_2[i]$ then $\mathsf{out} \leftarrow 1$	If $G(K_G, \mathbf{r}[i])\rvert_\zeta \neq \mathbf{s}_2[i]$ then $\mathsf{out} \leftarrow 1$
Return out	Return out
Procedure $\mathcal{O}(x)$	**Procedure** $\mathcal{O}(x)$
If $x \in \mathbf{x}$ then return $\perp$	If $x \in \mathbf{x}$ then return $\perp$
$(state, r) \leftarrow \mathcal{E}(state, \mathbf{x}, \mathbf{c}, x; w)$	$(state, r) \leftarrow \mathcal{E}(state, \mathbf{x}, \mathbf{c}, x; w)$
$\mathbf{r}[i] \leftarrow r$; $\mathbf{s}_2[i] \leftarrow x$; $i \leftarrow i + 1$	$\mathbf{r}[i] \leftarrow r$; $\mathbf{s}_2[i] \leftarrow x$; $i \leftarrow i + 1$
Return r	Return r
Procedure $\mathcal{I}(\mathcal{M})$	**Procedure** $\mathcal{I}(\mathcal{M})$
$m \leftarrow\$ \mathcal{M}(1^k)$; $r \leftarrow\$ \mathsf{GDom}(k)$	$r \leftarrow\$ \mathsf{GDom}(k)$; $s \leftarrow G(K_G, r)$
$s \leftarrow\$ \mathsf{GRng}(k)$; $s \leftarrow G(K_G, r)$; $x \leftarrow s \oplus m \Vert 0^\zeta$	$t \leftarrow r$; $y \leftarrow f(t)$
$t \leftarrow\$ \mathsf{GDom}(k)$; $y \leftarrow f(t)$	$\mathbf{x}[j] \leftarrow s\rvert_\zeta$; $\mathbf{c}[j] \leftarrow (s, y)$; $j \leftarrow j + 1$
$\mathbf{x}[j] \leftarrow s\rvert_\zeta$; $\mathbf{c}[j] \leftarrow (s, y)$; $j \leftarrow j + 1$	Return (s, c)
Return (s, c)	

Figure 76: **Games G_1-G_7 in the proof of Lemma 7.11.**

Summing up,

$$\mathbf{Adv}^{\eta\text{-ext2}_\zeta}_{\mathcal{G},\mathsf{OAEP}_{\mathsf{s-clear}},A,\mathsf{Ext}}(k) \leq \mathbf{Adv}^{\delta\text{-ext2}_\zeta}_{\mathcal{G},\mathcal{F},B,\mathsf{Ext}}(k) + 3p \cdot \mathbf{Adv}^{\text{xor-ind2}_\zeta}_{\mathcal{F},\mathcal{G},C}(k) + 2p \cdot \mathbf{Adv}^{\text{vprg}_\zeta}_{\mathcal{G},D}(k) \ .$$

This completes the proof.

Algorithm $C_{\ell,1}(f, K_G, x)$
$\mathbf{x} \leftarrow \varepsilon$; $st \leftarrow \varepsilon$; $z \leftarrow \varepsilon$; $\mathbf{c} \leftarrow \varepsilon$
$j \leftarrow 1$; $w \leftarrow\!\!\$\ \mathsf{Coins}(k)$
$K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$
$state \leftarrow (K_G, K_H, f)$
Run $A^{\mathcal{O}\text{SIM}(\cdot),\mathcal{I}\text{SIM}(\cdot)}(K_G, K_H, f; w)$
Return (st, z)

Procedure $\mathcal{O}\text{SIM}(v)$
If $v \in \mathbf{x}$ then return $\perp$
$(state, r) \leftarrow \mathcal{E}(state, \mathbf{x}, \mathbf{c}, v; w)$
Return r

Procedure $\mathcal{I}\text{SIM}(\mathcal{M})$
If $j < \ell$
  $m \leftarrow\!\!\$\ \mathcal{M}(1^k)$; $r \leftarrow\!\!\$\ \mathsf{GDom}(k)$
  $x' \leftarrow G(K_G, r)$; $s' \leftarrow x' \oplus m \| 0^\zeta$
  $t \leftarrow r$; $y \leftarrow f(t)$; $c \leftarrow (s', y)$
  $\mathbf{x}[j] \leftarrow x'|_\zeta$; $\mathbf{c}[j] \leftarrow c$; $j \leftarrow j + 1$
  Return (s', c)
If $j = \ell$
  $m \leftarrow\!\!\$\ \mathcal{M}(1^k)$; $s \leftarrow x \oplus m \| 0^\zeta$
  $z \leftarrow H(K_H, s)$; $st \leftarrow (K_G, K_H, f, w, s, x)$
  Halt A

Algorithm $C^{\mathcal{V}(\cdot)}_{\ell,2}(st, y_b)$
$i \leftarrow 1$; $j \leftarrow 1$
$\mathbf{x} \leftarrow \varepsilon$; $\mathbf{c} \leftarrow \varepsilon$; $\mathbf{r} \leftarrow \varepsilon$; $\mathbf{s}_2 \leftarrow \varepsilon$
$(K_G, K_H, f, w, s, x) \leftarrow st$
Run $A^{\mathcal{O}\text{SIM}(\cdot),\mathcal{I}\text{SIM}(\cdot)}(K_G, K_H, f; w)$
$b' \leftarrow 0$
For $i = 1$ to $|\mathbf{r}|$ do
  If $(\mathcal{V}(\mathbf{s}_2[i]) = 1 \wedge G(K_G, \mathbf{r}[i])|_\zeta \neq \mathbf{s}_2[i])$ then $b' \leftarrow 1$
Return b'

Procedure $\mathcal{O}\text{SIM}(v)$
If $v \in \mathbf{x}$ then return $\perp$
$(state, r) \leftarrow \mathcal{E}(state, \mathbf{x}, \mathbf{c}, v; w)$
$\mathbf{r}[i] \leftarrow r$; $\mathbf{s}_2[i] \leftarrow v$; $i \leftarrow i + 1$
Return r

Procedure $\mathcal{I}\text{SIM}(\mathcal{M})$
$m \leftarrow\!\!\$\ \mathcal{M}(1^k)$; $r \leftarrow\!\!\$\ \mathsf{GDom}(k)$
$x' \leftarrow G(K_G, r)$; $s' \leftarrow x' \oplus m \| 0^\zeta$
If $j < \ell$ then $t \leftarrow r$; $y \leftarrow f(t)$; $c \leftarrow (s', y)$
If $j = \ell$ then $s' \leftarrow s$; $c \leftarrow (s', y_b)$
If $j > \ell$ then $t \leftarrow H(K_H, s') \oplus r$; $y \leftarrow f(t)$; $c \leftarrow (s', y)$
$\mathbf{x}[j] \leftarrow x'|_\zeta$; $\mathbf{c}[j] \leftarrow c$; $j \leftarrow j + 1$
Return (s', c)

Figure 77: **Adversary C_ℓ in the proof of Lemma 7.11.**

Algorithm $D^{\mathcal{V}(\cdot)}_{\ell}(K_G, x)$
$i \leftarrow 1$; $j \leftarrow 1$; $b' \leftarrow 0$
$\mathbf{x} \leftarrow \varepsilon$; $\mathbf{c} \leftarrow \varepsilon$; $\mathbf{r} \leftarrow \varepsilon$; $\mathbf{s}_2 \leftarrow \varepsilon$
$w \leftarrow\!\!\$\ \mathsf{Coins}(k)$; $(f, f^{-1}) \leftarrow\!\!\$\ \mathsf{Kg}(1^k)$
$K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$; $state \leftarrow (K_G, K_H, f)$
Run $A^{\mathcal{O}\text{SIM}(\cdot),\mathcal{I}\text{SIM}(\cdot)}(K_G, K_H, f; w)$
For $i = 1$ to $|\mathbf{r}|$ do
  If $(\mathcal{V}(\mathbf{s}_2[i]) = 1 \wedge G(K_G, \mathbf{r}[i])|_\zeta \neq \mathbf{s}_2[i])$
    $b' \leftarrow 1$
Return b'

Procedure $\mathcal{O}\text{SIM}(v)$
If $v \in \mathbf{x}$ then return $\perp$
$(state, r) \leftarrow \mathcal{E}(state, \mathbf{x}, \mathbf{c}, v; w)$
$\mathbf{r}[i] \leftarrow r$; $\mathbf{s}_2[i] \leftarrow v$; $i \leftarrow i + 1$
Return r

Procedure $\mathcal{I}\text{SIM}(\mathcal{M})$
If $j < \ell$ then $x' \leftarrow\!\!\$\ \mathsf{GRng}(k)$
If $j = \ell$ then $x' \leftarrow x$
If $j > \ell$ then
  $r \leftarrow\!\!\$\ \mathsf{GDom}(k)$; $x' \leftarrow G(K_G, r)$
$m \leftarrow\!\!\$\ \mathcal{M}(1^k)$; $s' \leftarrow x' \oplus m \| 0^\zeta$
$t \leftarrow\!\!\$\ \mathsf{GDom}(k)$; $y \leftarrow f(t)$
$\mathbf{x}[j] \leftarrow x'|_\zeta$; $c \leftarrow (s', y)$
$\mathbf{c}[j] \leftarrow c$; $j \leftarrow j + 1$
Return (s', c)

Figure 78: **Adversary D_ℓ in the proof of Lemma 7.11.**

E.2 Proof of Lemma 7.12

Consider EXT1 adversary B_1 and EXT2 adversary B_2 in Figure 80. Let Ext_1 and Ext_2 be the extractors for adversaries B_1 and B_2, respectively.

Now, consider the PRG adversary D in Figure 81. Note that we have

$$\Pr\left[\,\text{PRG-REAL}^D_{\mathcal{G}} \Rightarrow 1\,\right] = \Pr\left[\,\text{VPRG-REAL}^A_{\mathcal{G}} \Rightarrow 1\,\right] - \mathbf{Adv}^{\delta\text{-ext1}_\zeta}_{\mathcal{G},B_1,\mathsf{Ext}_1}(k) \ .$$

Adversary $B^{\mathcal{O}(\cdot),\mathcal{I}(\cdot)}(K_G, f, K_H; w)$ Run $A^{\mathcal{O}\text{SIM}(\cdot),\mathcal{I}\text{SIM}(\cdot)}(K_G, f, K_H; w)$ **Procedure** $\mathcal{O}\text{SIM}(x)$ Return $\mathcal{O}(x)$	**Procedure** $\mathcal{I}\text{SIM}(\mathcal{M})$ $(x, y) \leftarrow \mathcal{I}(1^k)$ $c \leftarrow (x, y)$ Return (x, c)

Figure 79: **EXT2 adversary B in the proof of Lemma 7.11.**

Adversary $B_1^{\mathcal{O}(\cdot)}(K_G, aux; w)$ Run $A^{\mathcal{V}_\zeta \text{SIM}(\cdot)}(K_G, aux; w)$ **Procedure** $\mathcal{V}_\zeta\text{SIM}(x')$ $r \leftarrow \mathcal{O}(x)$ If $r = \perp$ then return 0 Return 1	**Adversary** $B_2^{\mathcal{O}(\cdot),\mathcal{I}(\cdot)}(K_G; w)$ $x \leftarrow \mathcal{I}(1^k)$ Run $A^{\mathcal{V}_\zeta \text{SIM}(\cdot)}(K_G, x; w)$ **Procedure** $\mathcal{V}_\zeta\text{SIM}(x')$ $r \leftarrow \mathcal{O}(x)$ If $r = \perp$ then return 0 Return 1

Figure 80: **Adversaries B_1, B_2 in the proof of Lemma 7.12.**

Moreover, we also know that

$$\Pr\left[\,\text{PRG-RAND}_{\mathcal{G}}^{D} \Rightarrow 1\,\right] = \Pr\left[\,\text{VPRG-RAND}_{\mathcal{G}}^{A} \Rightarrow 1\,\right] - \mathbf{Adv}^{\delta\text{-ext2}_\zeta}_{\mathcal{G},B_2,\mathsf{Ext}_2}(k) \ .$$

Summing up,

$$\mathbf{Adv}^{\text{vprg}_\zeta}_{\mathcal{G},A}(k) \leq \mathbf{Adv}^{\text{prg}}_{\mathcal{G},D}(k) + 2 \cdot \mathbf{Adv}^{\delta\text{-ext2}_\zeta}_{\mathcal{G},B,\mathsf{Ext}}(k) \ .$$

This completes the proof.

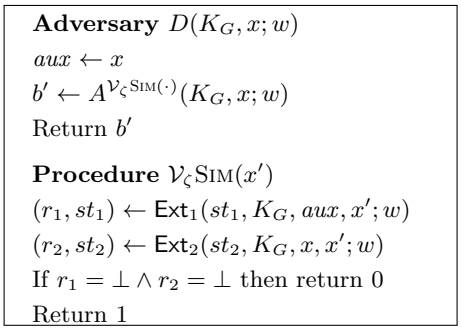

Figure 81: **PRG adversary D in the proof of Lemma 7.12.**