Instantiation of RO Model Transforms via Extractable Functions

Mohammad Zaheri[1]

July 28, 2020

Abstract

We show two new results about instantiability of the classical random-oracle-model encryption transforms for upgrading "weak" trapdoor permutations and encryption to "strong" chosen-ciphertext (CCA) secure encryption, namely the OAEP trapdoor permutation based (Bellare and Rogaway, EUROCRYPT 1994) and Fujasaki Okamoto (FO) hybrid-encryption (EUROCRYPT 1998) transforms:

- First, we propose a slight tweak to FO so that achieves the same goal in the RO model, but it is not "admissible" in the sense of Brzuska *et al.* (TCC 2015) and thus their uninstantiability result does not apply. We then show this modified transform is *fully instantiable* using extractable hash functions.
- Second, we show that OAEP is *partially instantiable* using extractability assumptions on the round function when trapdoor permutation is partially one-way. This improves the prior work by Cao *et al.* (PKC 2020) who showed weaker results. This shed light on "why" RSA-OAEP may be secure whereas there exists one-way trapdoor permutations for which the OAEP transform fails (Shoup, J. Cryptology 2002).

Keywords: Fujasaki-Okamoto Transform, RSA-OAEP, Random Oracle, Chosen-Ciphertext Security, Extractable Functions

[1] Dept. of Computer Science, Georgetown University, Email: mz394@georgetown.edu

Contents

1 Introduction **3**
1.1 Background and Goal . . . 3
1.2 High-Level Approach . . . 3
1.3 Our Results . . . 4
1.4 Further Related Work . . . 4

2 Preliminaries **4**
2.1 Notation and Conventions . . . 4
2.2 Encryption Schemes and Their Security . . . 5
2.3 Trapdoor Permutations and Their Security . . . 6
2.4 Function Families and Associated Security Notions . . . 7
2.5 The OAEP Framework . . . 8
2.6 The Fujisaki-Okamoto Transform . . . 9

3 Fujisaki-Okamoto Transform Instantiation **9**

4 RSA-OAEP Instantiation **11**

1 Introduction

In this paper, we show new partial and full instantiations under chosen-ciphertext attack (CCA) for the Fujasaki-Okamoto [22] and OAEP [6] transforms. This helps explain why there are no attacks on these transforms despite the existence of "uninstantiable" RO model schemes. We now discuss some background and motivation before an overview of our results.

1.1 Background and Goal

The random oracle (RO) model [5] is a popular paradigm for designing practical cryptographic schemes. In this model, we design and analysis the security of scheme assuming all parties have access to one or more oracles that implement independent random functions (called the ROs). We then "instantiate" these ROs using cryptographic hash functions. The hope is that the instantiated scheme remains secure. However, Canetti *et al.* [16] show that this is false in a strong sense. In their work, they give schemes that are secure in the RO model but are insecure when instantiated with *any* real-world functions. We call these schemes, "uninstantiable."

RO model Transforms. Questions of uninstantiability are particularly concern the use of *transforms* (compilers that take one or more "base schemes" and output a "target scheme" that uses ROs) in the RO model. We refer to a transform as uninstantiable if for any standard-model hash functions replacing the ROs there exist secure base schemes such that the corresponding target scheme is insecure.

In this paper, we are concerned with instantiability of RO model transforms that output a (public-key) encryption scheme, particularly the classical Fujasaki-Okamoto (FO) hybrid-encryption transform [22] and OAEP trapdoor-permutation-based transform [6]. We start by recalling recall about how these two classical encryption scheme transforms work and what is known about them.

Fujasaki-Okamoto. This transform takes a public-key encryption scheme and a symmetric-key encryption scheme, and produces a new public-key encryption scheme as follow:

$$\mathcal{E}^{\mathsf{hy}}_{pk}(m;r) = \mathcal{E}^{\mathsf{asy}}_{pk}(r;\mathsf{H}(r))\|\mathcal{E}^{\mathsf{sy}}_{K}(m) \quad \text{where} \quad K = \mathsf{G}(r) \ .$$

Hofheinz *et al.* [27] show that the resulting public-key encryption scheme $\mathcal{E}^{\mathsf{hy}}$ is IND-CCA secure for public-key schemes $\mathcal{E}^{\mathsf{asy}}$ that are one-way CPA and IND-CCA symmetric-key encryption schemes $\mathcal{E}^{\mathsf{sy}}$ when H, G are ROs. Unfortunately, FO was shown uninstantiable by Brzuska *et al.* [11]. They showed uninstantiability of all "admissible" such encryption transforms and that FO is admissible *regardless* of the class of symmetric-key schemes considered.

OAEP. This transform takes a trapdoor permutation (TDP) $\mathcal{F}$ and produces a public-key encryption scheme whose public key is an instance f of the TDP. More specifically, the resulting transform is as follow:

$$\mathcal{E}^{\mathsf{OAEP}}_{f}(m;r) = f(s\|t) \quad \text{where} \quad s = \mathsf{G}(r)\oplus m\|0^{\zeta} \quad \text{and} \quad t = \mathsf{H}(s)\oplus r \ .$$

Shoup [31] showed that it cannot be secure for *every* one-way trapdoor permutation. But this result does not apply to practical TDPs, let alone the commonly used RSA TDP. Further, there are black-box impossibility results imply that one either has to use non-blackbox assumptions on the hash functions or on the TDP [29]. Moreover, Cao *et al.* [18] show partial instantiation result for OAEP transform under mild assumptions on G, H when TDP satisfies the notions of "second-input extractability" (SIE) and "common-input extractability" (CIE). Barthe *et al.* [2] show that these extractability assumptions hold for small-exponent RSA (e = 3).

The main question that left open by prior work is that; Are there standard model hash functions that suffice to instantiate OAEP and FO (under IND-CCA2) for classes of "practical" base schemes? This main question is the starting point for our work.

1.2 High-Level Approach

The high level idea is replacing ROs with extractable functions to instantiate FO and OAEP transforms. Extractable functions are first introduced in [14, 15, 20, 8]. Later, Cao *et al.* [18] introduced a hierarchy of extractability notions, called EXT-RO, EXT0, EXT1, EXT2. Intuitively, extractability of a function formalizes the idea that an adversary that produces a point in the image must "know" a corresponding preimage, as there being a non-blackbox extractor that produces one.

Previously, Cao *et al.* [18] show how to use extractable functions to fully instantiate the variants of RSA-OAEP. We built on their work and show instantiation results on FO and OAEP transforms.

1.3 Our Results

Results on Fujasaki-Okamoto. The Fujasaki-Okamoto transform takes a chosen plaintext attack secure public-key encryption scheme and lifts it to chosen ciphertext security by using it in a hybrid construction with a symmetric. We consider a slightly modified FO transform.

$$\mathcal{E}^{\mathsf{hy}}_{pk}(m;r) = \mathcal{E}^{\mathsf{asy}}_{pk}(f(r);\mathsf{H}(r)) \| \mathcal{E}^{\mathsf{sy}}_{K}(m) \quad \text{where} \quad K = \mathsf{G}(r) \ ,$$

where f is a trapdoor permutation. Observe that compare to original FO, we encrypt $f(r)$ instead of r. We show this modified transform is fully instantiable under suitable assumptions. We assume the public-key encryption scheme is uniquely randomness recovering and the symmetric-key encryption is AE. Then to instantiate H, G we use extractable functions and one-wayness extractor. Note that we we remove the OW-CPA assumption on public-key encryption since we encrypt $f(r)$ instead of r.

We sketch the proof that the instantiated scheme is IND-CCA secure. Let $c^* = (c_1^*, c_2^*)$ be the challenge ciphertext and $c = (c_1, c_2)$ be the decryption query made by IND-CCA adversary. We first show that if H is suitably extractable then there is an extractor that on input $c_1 = \mathcal{E}^{\mathsf{asy}}_{pk}(f(r);\mathsf{H}(r))$ can simply recover r. We then use r to decrypt $c_2 = \mathcal{E}^{\mathsf{sy}}_{K}(m)$ and recover m. We use this extractor to answer to the decryption queries made by IND-CCA adversary. We note that the extractor fails to recover r if $c_1 = c_1^*$. However, for this to happens we show that adversary needs to come up with $c_2 = \mathcal{E}^{\mathsf{sy}}_{K}(m)$ where $m \neq m^*$. This happens with negligible probability since symmetric-key encryption is AE. Next we show that (c_1, c_2) looks random since G is one-wayness extractor and symmetric-key encryption is AE.

Results on OAEP. We show that OAEP is partially instantiable under suitable assumptions. Cao *et. al* [18] show how to partially instantiate either G or H for RSA-OAEP under IND-CCA. Their results require RSA to be second input and common input extractable. These algebraic properties proven to hold for RSA with small exponent (*i.e.* $e = 3$). However in practice RSA is used with much larger exponent. We show how to trade the extractability assumption on RSA with extractability assumption on round function G to partially instantiate RSA-OAEP even for the large exponent e. In particular, we show RSA-OAEP is partially instantiable when G is extractable, collision resistance pseudorandom generator while H is a RO. Note that we only require RSA to be partially one-way.

We sketch the proof that the instantiated scheme is IND-CCA secure. Let c^* be the challenge ciphertext and c be the decryption query made by IND-CCA adversary. We first show that if G is suitably extractable then there is an extractor that can recover m. We use this extractor to answer to the decryption queries made by IND-CCA adversary. We note that the extractor fails to recover m if the most significant bits of preimage c and c^* are equal. However, for this to happens we show that adversary needs to create a collision on G. Thus, this happens with negligible probability since G is collision resistance. Next we show that c looks random since G is PRG and RSA is partially one-way.

1.4 Further Related Work

There are several candidates proposed to replace ROs including correlation intractability [16, 13], perfect one-wayness [12, 17, 21], non-malleability [9, 1], seed incompressibility [26], and universal computational extraction (UCE) [3, 10, 4].

2 Preliminaries

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 $\Pr[\,A(x) = y :\ x \leftarrow_{\$} X\,]$ 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,

$$
\begin{array}{|l|}
\hline
\textbf{Game } \text{IND-CCA}_{\mathsf{SE}}^{A}(k) \\
b \leftarrow_{\$} \{0,1\}\ ;\ K \leftarrow_{\$} \mathcal{K}(1^k) \\
(\mathcal{M}_0, \mathcal{M}_1, \mathit{state}) \leftarrow_{\$} A_1^{\mathcal{D}_K(\cdot)}(1^k) \\
m_b \leftarrow_{\$} \mathcal{M}_b(1^k) \\
c \leftarrow_{\$} \mathcal{E}_K(m_b) \\
d \leftarrow_{\$} A_2^{\mathcal{D}_K(\cdot)}(c, \mathit{state}) \\
\text{Return } (b = d) \\
\hline
\end{array}
$$

Figure 1: **Game to define** IND-CCA **security for private-key encryption.**

$$
\begin{array}{|l|}
\hline
\textbf{Game } \text{INT-CTXT}_{\mathsf{SE}}^{A}(k) \\
K \leftarrow_{\$} \mathcal{K}(1^k) \\
c^* \leftarrow_{\$} A^{\mathcal{E}_K(\cdot)}(1^k) \\
\text{If } \mathcal{D}_K(c^*) \neq \bot \text{ then return } 1 \\
\text{Return } 0 \\
\hline
\end{array}
$$

Figure 2: **Game to define** INT-CTXT **security for private-key encryption.**

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 $s|_\ell$ the ℓ least significant bits (LSB) of s and $s|^\ell$ the ℓ most significant bits (MSB) of s, for $1 \leq \ell \leq |s|$. 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 $1 \leq i \leq |\mathbf{x}|$. 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 $1 \leq i \leq |\mathbf{x}|$.

Unpredictable distribution. We call distribution ensemble $D = \{D_k\}_{k \in \mathbb{N}}$, on pairs of strings (Z_k, X_k), unpredictable if for every PPT algorithm A, we have

$$\Pr\left[\, A(1^k, z) = x :\ (x, z) \leftarrow_{\$} D_k \,\right] \ ,$$

is negligible in k.

2.2 Encryption Schemes and Their Security

Private-key encryption. A *private-key encryption scheme* SE with message space Msg is a tuple of algorithms $(\mathcal{K}, \mathcal{E}, \mathcal{D})$. The key-generation algorithm $\mathcal{K}$ on input 1^k outputs a private key K. The encryption algorithm $\mathcal{E}$ on inputs K and a message $m \in \mathsf{Msg}(1^k)$ outputs a ciphertext $c \in \mathsf{Ctxt}(1^k)$. The deterministic decryption algorithm $\mathcal{D}$ on inputs K and ciphertext c outputs a message m or $\bot$. We require that for all $K \in [\mathcal{K}(1^k)]$ and all $m \in \mathsf{Msg}(1^k)$, $\mathcal{D}_K((\mathcal{E}_K(m)) = m$ with probability 1.

Security of private-key encryption. Let $\mathsf{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a private key encryption scheme and $A = (A_1, A_2)$ be an adversary. Let $\mathcal{M}$ be a PPT algorithm that takes inputs 1^k to return a message $m \in \mathsf{Msg}(1^k)$. We associate the experiment in Figure 1 for every $k \in \mathbb{N}$. Define the *ind-cca advantage* of A against SE as

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

We note that A_2 is not allowed to ask $\mathcal{D}$ to decrypt c. We say SE is *secure under chosen-ciphertext attack* (IND-CCA) if $\mathbf{Adv}^{\text{ind-cca}}_{\mathsf{SE},A}(k)$ is negligible in k for all PPT A.

Integrity of private-key encryption. Let $\mathsf{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a private key encryption scheme, and A be an adversary. We associate the experiment in Figure 2 for every $k \in \mathbb{N}$. Define the *int-ctxt advantage* of A against SE as

$$\mathbf{Adv}^{\text{int-ctxt}}_{\mathsf{SE},A}(k) = \Pr\left[\, \text{INT-CTXT}^{A}_{\mathsf{SE}}(k) \Rightarrow 1 \,\right] \ .$$

We say that SE is secure under INT-CTXT, if $\mathbf{Adv}^{\text{int-ctxt}}_{\mathsf{SE},A}(k)$ is negligible in k for all PPT A.

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

$$
\boxed{\begin{array}{l}
\textbf{Game } \text{IND-ATK}^{A}_{\mathsf{PKE}}(k) \\
b \leftarrow_{\$} \{0,1\}\ ;\ (pk, sk) \leftarrow_{\$} \mathsf{Kg}(1^k) \\
(\mathcal{M}_0, \mathcal{M}_1, \mathit{state}) \leftarrow_{\$} A_1^{\mathcal{O}(\cdot)}(1^k, pk) \\
m_b \leftarrow_{\$} \mathcal{M}_b(1^k, pk) \\
c \leftarrow_{\$} \mathsf{Enc}(pk, m_b) \\
d \leftarrow_{\$} A_2^{\mathcal{O}(\cdot)}(pk, c, \mathit{state}) \\
\text{Return } (b = d)
\end{array}}
$$

Figure 3: **Game to define** IND-ATK **security for public-key encryption.**

key sk. The encryption algorithm Enc on inputs pk and a message $m \in \mathsf{Msg}(1^k)$ outputs a ciphertext c. The 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 [25, 30]. 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, CCA} we associate the experiment in Figure 3 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 $\mathcal{O}(\cdot) = \varepsilon$. 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 = cca, then $\mathcal{O}(\cdot) = \mathsf{Dec}(sk, \cdot)$. Note that adversary A_2 is not allowed to ask $\mathcal{O}$ to decrypt c. We say that PKE is secure under adaptive chosen-ciphertext attack or IND-CCA, if $\mathbf{Adv}^{\text{ind-cca}}_{\mathsf{PKE},A}(k)$ is negligible in k for all PPT A.

Randomness recovery [19]. Let $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a public key encryption. We say PKE is *uniquely randomness recovering* if there exist a PT randomness recovery algorithm Rec such that on input a secret key sk and ciphertext c outputs a randomness r. We require that for all $(pk, sk) \in [\mathsf{Kg}(1^k)]$, all randomness r and all $m \in \mathsf{Msg}(1^k)$, $\mathsf{Rec}(sk, (\mathsf{Enc}(pk, m; r)) = r$ with probability 1.

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 $x \in \mathsf{TDom}(k)$, $f^{-1}(f(x)) = x$.

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:

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

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 *partial one-way* with respect to ℓ-most significant bits of the challenge input (ℓ-POW) if for every PPT inverter I:

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

is negligible in k. It is shown in [23] that for RSA one-wayness implies partial one-wayness but the reduction is lossy.

Game $\mathrm{EXT2}^{A,\mathsf{Ext},z}_{\mathcal{F},D}(K_F, r)$	Procedure $\mathcal{O}(y)$
$i \leftarrow 1$; $j \leftarrow 1$	If $y \in \mathbf{f}$ then return $\perp$
$state \leftarrow \varepsilon$	$(state, x) \leftarrow \mathsf{Ext}(state, K_F, z, \mathbf{f}, \mathbf{hint}, y; r)$
$\mathbf{x} \leftarrow \varepsilon$; $\mathbf{y} \leftarrow \varepsilon$	$\mathbf{x}[i] \leftarrow x$; $\mathbf{y}[i] \leftarrow y$; $i \leftarrow i+1$
$\mathbf{f} \leftarrow \varepsilon$; $\mathbf{hint} \leftarrow \varepsilon$	Return x
Run $A^{\mathcal{O}(\cdot),\mathcal{I}(\cdot)}(K_F, z; r)$	**Procedure** $\mathcal{I}(1^k)$
Return $(\mathbf{x}, \mathbf{y})$	$(\mathsf{hint}, v) \leftarrow_{\$} D(1^k)$; $f \leftarrow F(K_F, v)$
	$\mathbf{f}[j] \leftarrow f$; $\mathbf{hint}[j] \leftarrow \mathsf{hint}$; $j \leftarrow j+1$
	Return (f, hint)

Figure 4: **Game to define EXT2 security.**

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 $\mathcal{F} \colon \mathcal{K}_F \times \mathsf{F.Dom} \to \mathsf{F.Rng}$.

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

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

is negligible in k.

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

$$\mathbf{Adv}^{\mathrm{n\text{-}cr}}_{\mathcal{F},A}(k) = \Pr_{K_F \leftarrow_{\$} \mathcal{K}_F(1^k)}\left[\begin{array}{c} (x_1, x_2) \leftarrow A(K_F) \\ x_1, x_2 \in \mathsf{F.Dom}(k) \end{array} \wedge \begin{array}{c} F(K_F, x_1)|^{\ell} = F(K_F, x_2)|^{\ell} \\ x_1 \neq x_2 \end{array}\right] ,$$

is negligible in k.

Hardcore functions. We recall a notion of hardcore functions in [24]. 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. We say that $\mathcal{H}$ is a *hardcore function* for the trapdoor permutation family $\mathcal{F}$ if for every PPT adversary A,

$$\mathbf{Adv}^{\mathrm{hcf}}_{\mathcal{F},\mathcal{H},A}(k) = \Pr\left[\, A(K_H, f, f(x), H(K_H, x)) = 1 \,\right] - \Pr\left[\, A(K_H, f, f(x), U) = 1 \,\right] ,$$

is negligible in k, where $K_H \leftarrow_{\$} \mathcal{K}_H(1^k)$, $f \leftarrow_{\$} \mathsf{Kg}(1^k)$, x is chosen uniformly random from domain $\mathsf{TDom}(k)$, and $U \leftarrow_{\$} \mathsf{HRng}(k)$.

Pseudorandom generators. Let $\mathcal{F} \colon \mathcal{K}_F \times \mathsf{F.Dom} \to \mathsf{F.Rng}$ be a function family. We say that $\mathcal{F}$ is a *pseudorandom generator* (PRG) if for every PPT adversary A,

$$\mathbf{Adv}^{\mathrm{prg}}_{\mathcal{F},A}(k) = \Pr\left[\, A(K_F, F(K_F, x)) = 1 \,\right] - \Pr\left[\, A(K_F, U) = 1 \,\right] .$$

is negligible in k, where $K_F \leftarrow_{\$} \mathcal{K}_F(1^k)$, $x \leftarrow_{\$} \mathsf{F.Dom}(k)$, and $U \leftarrow_{\$} \mathsf{F.Rng}(k)$.

One-wayness extractors. Let $\mathcal{F} \colon \mathcal{K}_F \times \mathsf{F.Dom} \to \mathsf{F.Rng}$ be a function family. We say $\mathcal{F}$ is a *one-wayness extractor* [28] if for any PPT adversary A and any unpredictable distribution D we have

$$\mathbf{Adv}^{\mathsf{cdist}}_{\mathcal{F},A,D} = \Pr\left[\, A(K_F, z, F(K_F, x)) = 1 \,\right] - \Pr\left[\, A(K_F, z, U) = 1 \,\right] ,$$

is negligible in k, where $K_F \leftarrow_{\$} \mathcal{K}_F(1^k)$, $(z, x) \leftarrow_{\$} D_k$, and $U \leftarrow_{\$} \mathsf{F.Rng}(k)$.

Extractable functions. Intuitively, extractability of a function families formalizes the idea that an adversary that produces an image point must "know" a corresponding preimage, as there being a non-blackbox extractor

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

Figure 5: **Padding based encryption scheme** $\mathsf{PAD}[\mathcal{F}] = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$**.**

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

Figure 6: **OAEP padding scheme** $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}]$**.**

that recovers the preimage. Cao *et al.* in [18], defined a hierarchy of EXT for function families, namely EXT0, EXT1, and EXT2, which shown to be useful for instantiating RSA-OAEP. Here we recall the EXT2 notion.

Let η be integer parameters. Let $\mathcal{F} : \mathcal{K}_F \times \mathsf{F.Dom} \to \mathsf{F.Rng}$ be a hash function family and $D = \{D_k\}_{k\in\mathbb{N}}$ be an unpredictable distribution on domain $\mathsf{F.Dom}$. To adversary A and extractor Ext, we associate the experiment in Figure 4, for every $k \in \mathbb{N}$. We say $\mathcal{F}$ is η-EXT2 if for any PPT adversary A with coin space Coins, and any unpredictable distribution D, there exists a stateful extractor Ext such that, for any key independent auxiliary input $z \in \{0,1\}^\eta$:

$$\mathbf{Adv}^{\eta\text{-ext2}}_{\mathcal{F},D,A,\mathsf{Ext},z}(k) = \Pr_{\substack{K_F \leftarrow\$\ \mathcal{K}_F(1^k) \\ r \leftarrow\$\ \mathsf{Coins}(k)}}\left[\begin{array}{c}(\mathbf{x},\mathbf{y}) \leftarrow \mathrm{EXT2}^{A,\mathsf{Ext},z}_{\mathcal{F},D}(K_F, r) \\ \exists i, \exists x : F(K_F, x) = \mathbf{y}[i] \wedge F(K_F, \mathbf{x}[i]) \neq \mathbf{y}[i]\end{array}\right],$$

is negligible in k. The adversary is not allowed to query $y \in \mathbf{f}$ for extract oracle $\mathcal{O}$. We define advantage of A to be $\mathbf{Adv}^{\eta\text{-ext2}}_{\mathcal{F},D,A,\mathsf{Ext}}(k) = \max_{z\in\{0,1\}^\eta} \mathbf{Adv}^{\eta\text{-ext2}}_{\mathcal{F},D,A,\mathsf{Ext},z}(k)$.

We also extend the EXT2 notion to the case where the adversary only outputs ζ-bits of the image. We often write (η, ζ)-EXT2 for the function families that are extractable for such adversaries.

2.5 The OAEP Framework

Padding scheme. We define a general notion of padding scheme following [6, 29]. 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 $\mathcal{F}$ be a TDP with domain $\{0,1\}^\nu$. Let PAD be a padding transform from domain $\{0,1\}^{\mu+\rho}$ to range $\{0,1\}^\nu$. The associated *padding-based encryption scheme* is a triple of algorithms $\mathsf{PAD}[\mathcal{F}] = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ defined in Figure 5.

OAEP padding scheme. We recall the OAEP padding scheme [6]. 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 $K_G \leftarrow\!\!\$\ \mathcal{K}_G(1^k)$, $K_H \leftarrow\!\!\$\ \mathcal{K}_H(1^k)$, and $\mathsf{OAEP}, \mathsf{OAEP}^{-1}$ are as defined in Figure 6.

OAEP encryption scheme. We denote by $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ the OAEP-based encryption scheme $\mathcal{F}$-OAEP with $n = \nu$. We typically think of $\mathcal{F}$ as RSA, and all our results apply to this case under suitable assumptions.

$\mathsf{FO.Kg}(1^k)$	$\mathsf{FO.Enc}(pk, m; r)$	$\mathsf{FO.Dec}(sk, c)$
$(pk', sk') \leftarrow_{\$} \mathsf{Kg}(1^k)$	$y \leftarrow H(K_H, r)$	$r \leftarrow \mathsf{Dec}(sk', c_1)$
$K_H \leftarrow_{\$} \mathcal{K}_H(1^k)$	$c_1 \leftarrow \mathsf{Enc}(pk', r; y)$	$K \leftarrow G(K_G, r)$
$K_G \leftarrow_{\$} \mathcal{K}_G(1^k)$	$K \leftarrow G(K_G, r)$	$m \leftarrow \mathcal{D}_K(c_2)$
$pk \leftarrow (pk', K_H, K_G)$	$c_2 \leftarrow \mathcal{E}_K(m)$	Return m
$sk \leftarrow (sk', K_H, K_G)$	$c \leftarrow (c_1, c_2)$	
Return (pk, sk)	Return c	

Figure 7: **FO transform** $\mathsf{FO}_{\mathcal{H},\mathcal{G}}[\mathsf{PKE}, \mathsf{SE}] = (\mathsf{FO.Kg}, \mathsf{FO.Enc}, \mathsf{FO.Dec})$.

$\overline{\mathsf{FO}}.\mathsf{Kg}(1^k)$	$\overline{\mathsf{FO}}.\mathsf{Enc}(pk, m; r)$	$\overline{\mathsf{FO}}.\mathsf{Dec}(sk, c)$
$(pk', sk') \leftarrow_{\$} \mathsf{Kg}(1^k)$	$h \leftarrow H(K_H, r)$	$y \leftarrow \mathsf{Dec}(sk', c_1)$
$(f, f^{-1}) \leftarrow_{\$} \mathsf{Kg}(1^k)$	$c_1 \leftarrow \mathsf{Enc}(pk', f(r); h)$	$r \leftarrow f^{-1}(y)$
$K_H \leftarrow_{\$} \mathcal{K}_H(1^k)$	$K \leftarrow G(K_G, r)$	$K \leftarrow G(K_G, r)$
$K_G \leftarrow_{\$} \mathcal{K}_G(1^k)$	$c_2 \leftarrow \mathcal{E}_K(m)$	$m \leftarrow \mathcal{D}_K(c_2)$
$pk \leftarrow (pk', f, K_H, K_G)$	$c \leftarrow (c_1, c_2)$	Return m
$sk \leftarrow (sk', f^{-1}, K_H, K_G)$	Return c	
Return (pk, sk)		

Figure 8: **Our new transform** $\overline{\mathsf{FO}}_{\mathcal{F},\mathcal{H},\mathcal{G}}[\mathsf{PKE}, \mathsf{SE}] = (\overline{\mathsf{FO}}.\mathsf{Dec}, \overline{\mathsf{FO}}.\mathsf{Enc}, \overline{\mathsf{FO}}.\mathsf{Dec})$.

2.6 The Fujisaki-Okamoto Transform

The Fujisaki-Okamoto (FO) transformation [22] is a technique to convert weak public key encryption schemes, e.g., IND-CPA secure into strong ones which resist chosen ciphertext attacks (i.e., IND-CCA secure). Let $\mathsf{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a private-key encryption and $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a public-key encryption schemes. Moreover, let $\mathcal{H} \colon \mathcal{K}_H \times \mathsf{HDom} \to \mathsf{HRng}$ and $\mathcal{G} \colon \mathcal{K}_G \times \mathsf{GDom} \to \mathsf{GRng}$ be function families. We define FO transform $\mathsf{FO}_{\mathcal{H},\mathcal{G}}[\mathsf{PKE}, \mathsf{SE}] = (\mathsf{FO.Kg}, \mathsf{FO.Enc}, \mathsf{FO.Dec})$ in Figure 7.

3 Fujisaki-Okamoto Transform Instantiation

In this section, we slightly change the original FO transform and give a new transform which we call $\overline{\mathsf{FO}}$. Next we instantiate the new transform $\overline{\mathsf{FO}}$ using extractable functions. Let $\mathsf{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a private-key encryption, $\mathsf{PKE} = (\mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a public-key encryption schemes, and $\mathcal{F} = (\mathsf{Kg}, \mathsf{Eval}, \mathsf{Inv})$ be a trapdoor permutation family. Moreover, let $\mathcal{H} \colon \mathcal{K}_H \times \mathsf{HDom} \to \mathsf{HRng}$ and $\mathcal{G} \colon \mathcal{K}_G \times \mathsf{GDom} \to \mathsf{GRng}$ be function families. We define $\overline{\mathsf{FO}}_{\mathcal{F},\mathcal{H},\mathcal{G}}[\mathsf{PKE}, \mathsf{SE}] = (\overline{\mathsf{FO}}.\mathsf{Dec}, \overline{\mathsf{FO}}.\mathsf{Enc}, \overline{\mathsf{FO}}.\mathsf{Dec})$ in Figure 8.

Theorem 3.1 *Assuming $\mathcal{F}$ is a one-way trapdoor permutation family, $\mathcal{H}$ is a hardcore function for $\mathcal{F}$ and η-EXT2, and $\mathcal{G}$ is one-wayness extractor. Moreover, assuming* PKE *is uniquely randomness recovering, and* SE *is IND-CPA and INT-CTXT secure. Then* $\overline{\mathsf{FO}}$ *defined as above is IND-CCA secure.*

Proof: We prove security through a sequence of games. Consider games G_1–G_3 in Figure 9. We now explain the game chain.

Game G_1: Game G_1 is the standard indistinguishability chosen ciphertext (IND-CCA) game. Thus, we have that $\mathbf{Adv}^{\text{ind-cca}}_{\overline{\mathsf{FO}},A}(k) = 2 \cdot \Pr[\, G_1 \Rightarrow 1 \,] - 1$ for any PPT adversary A.

Game G_2: Game G_2 is similar to game G_1 except that we change the decryption oracle as follows. We first run randomness recovery algorithm Rec on inputs c_1 and secret key sk' to obtain h. Then we use the extractor for the hash function family $\mathcal{H}$ to extract the randomness r and decrypt c_2 using symmetric key $G(K_G, r)$. Consider EXT2 adversary B in Figure 10. Let Ext be an extractor for adversary B. We note that hint given to adversary B by image oracle $\mathcal{I}$ is uninvertible, since $\mathcal{G}$ is a one-wayness extractor and $\mathcal{F}$ is one-way.

Let $c_i = (c_{i,1}, c_{i,2})$ be the i-th decryption query that adversary A makes, where $c_{i,1} = \mathsf{Enc}(pk', f(r_i); h_i)$ and $c_{i,2} = \mathcal{E}_{G(K_G, r_i)}(m)$. Note that for all queries $c_i \neq c^*$ if we have $h_i = h^*$ then extractor Ext fails. Otherwise extractor Ext can successfully extract r_i. Thus, we need to bound the probability of $h_i = h^*$ for any $i \in [p]$ where p is the number of decryption queries that adversary A makes. Note that we have

$$\Pr[\, h_i = h^* \,] = \Pr[\, h_i = h^* \wedge f(r_i) = f(r^*) \,] + \Pr[\, h_i = h^* \wedge f(r_i) \neq f(r^*) \,]$$

Games $G_1(k)$	**Games** $G_2(k)$, $\boxed{G_3(k)}$
$(pk', sk') \leftarrow_{\$} \mathsf{Kg}(1^k)$; $(f, f^{-1}) \leftarrow_{\$} \mathsf{Kg}(1^k)$	$(pk', sk') \leftarrow_{\$} \mathsf{Kg}(1^k)$; $(f, f^{-1}) \leftarrow_{\$} \mathsf{Kg}(1^k)$
$K_H \leftarrow_{\$} \mathcal{K}_H(1^k)$; $K_G \leftarrow_{\$} \mathcal{K}_G(1^k)$	$K_H \leftarrow_{\$} \mathcal{K}_H(1^k)$; $K_G \leftarrow_{\$} \mathcal{K}_G(1^k)$
$pk \leftarrow (pk', f, K_H, K_G)$	$pk \leftarrow (pk', f, K_H, K_G)$; $\mathsf{coin} \leftarrow_{\$} \mathsf{Coins}$
$(\mathcal{M}_0, \mathcal{M}_1, st) \leftarrow_{\$} A_1^{\mathsf{Dec}(\cdot)}(1^k, pk)$	$(\mathcal{M}_0, \mathcal{M}_1, st) \leftarrow A_1^{\mathsf{Dec}(\cdot)}(1^k, pk; \mathsf{coin})$
$b \leftarrow_{\$} \{0,1\}$; $m_b \leftarrow_{\$} \mathcal{M}_b(1^k, pk)$	$b \leftarrow_{\$} \{0,1\}$; $m_b \leftarrow \mathcal{M}_b(1^k, pk; \mathsf{coin})$
$r^* \leftarrow_{\$} \mathsf{HDom}(k)$; $c_1^* \leftarrow \mathsf{Enc}(pk', f(r^*); h^*)$	$r^* \leftarrow_{\$} \mathsf{HDom}(k)$; $c_1^* \leftarrow \mathsf{Enc}(pk', f(r^*); h^*)$
$K^* \leftarrow G(K_G, r^*)$; $c_2^* \leftarrow_{\$} \mathcal{E}_{K^*}(m_b)$	$K^* \leftarrow G(K_G, r^*)$; $\boxed{K^* \leftarrow_{\$} \mathsf{GRng}(k)}$
$d \leftarrow_{\$} A_2^{\mathsf{Dec}(\cdot)}(st, (c_1^*, c_2^*))$	$c_2^* \leftarrow_{\$} \mathcal{E}_{K^*}(m_b)$; $d \leftarrow A_2^{\mathsf{Dec}(\cdot)}(st, (c_1^*, c_2^*); \mathsf{coin})$
Return $(b = d)$	Return $(b = d)$

Procedure $\mathsf{Dec}(c)$ // of games G_2, G_3

$\mathsf{hint} \leftarrow (f, K_G, f(r^*), K^*)$; $aux \leftarrow (b, pk', sk')$

$(c_1, c_2) \leftarrow c$; $h \leftarrow \mathsf{Rec}(sk', c_1)$

$r \leftarrow \mathsf{Ext}(K_H, aux, \mathsf{coin}, \mathsf{hint}, h^*, h)$

$K \leftarrow G(K_G, r)$; $m \leftarrow \mathcal{D}_K(c_2)$

Return m

Figure 9: **Games G_1–G_3 in the proof of Theorem 3.1.**

Adversary $B^{\mathcal{O},\mathcal{I}}(K_H, aux; \mathsf{coin})$	**Procedure** $\mathsf{Dec}(c)$
$(b, pk', sk') \leftarrow aux$; $(\mathsf{hint}, h^*) \leftarrow_{\$} \mathcal{I}(1^k)$	$(c_1, c_2) \leftarrow c$
$(f, K_G, f(r^*), G(K_G, r^*)) \leftarrow \mathsf{hint}$	$h \leftarrow \mathsf{Rec}(sk', c_1)$
$pk \leftarrow (pk', f, K_H, K_G)$	$r \leftarrow \mathcal{O}(h)$
$(\mathcal{M}_0, \mathcal{M}_1, st) \leftarrow A_1^{\mathsf{Dec}(\cdot)}(pk; \mathsf{coin})$	$K \leftarrow G(K_G, r)$
$m_b \leftarrow \mathcal{M}_b(1^k, pk; \mathsf{coin})$	$m \leftarrow \mathcal{D}_K(c_2)$
$c_1^* \leftarrow \mathsf{Enc}(pk', f(r^*); h^*)$	Return m
$K^* \leftarrow G(K_G, r^*)$; $c_2^* \leftarrow_{\$} \mathcal{E}_{K^*}(m_b)$	
Run $A_2^{\mathsf{Dec}(\cdot)}(st, (c_1^*, c_2^*); \mathsf{coin})$	

Figure 10: **Adversary B in the proof of Theorem 3.1.**

Observe that when adversary A makes a decryption query $c_i \neq c^*$ such that $h_i = h^*$ and $f(r_i) = f(r^*)$, we are able to construct INT-CTXT adversary B_1 attacking symmetric key encryption SE. Thus, we obtain $\Pr[\, h_i = h^* \wedge f(r_i) = f(r^*) \,] \leq p \cdot \mathbf{Adv}^{\text{int-ctxt}}_{\mathsf{SE},B_1}(k)$. On the other hand, when adversary A makes a decryption query $c_i \neq c^*$ such that $h_i = h^*$ and $f(r_i) \neq f(r^*)$, we are able to construct adversary B_2 attacking function family $\mathcal{H}$ that can successfully find collisions. Hence, we obtain that $\Pr[\, h_i = h^* \wedge f(r_i) \neq f(r^*) \,] \leq \mathbf{Adv}^{\text{cr}}_{\mathcal{H},B_2}(k)$. Summing up, we obtain that $\Pr[\, G_1 \Rightarrow 1 \,] - \Pr[\, G_2 \Rightarrow 1 \,] \leq p \cdot \mathbf{Adv}^{\text{int-ctxt}}_{\mathsf{SE},B_1}(k) + \mathbf{Adv}^{\text{cr}}_{\mathcal{H},B_2}(k) + \mathbf{Adv}^{\text{ext2}}_{\mathcal{H},B,\mathsf{Ext}}(k)$.

Game G_3: Game G_3 is similar to game G_2 except that K^* is chosen at random in $\mathsf{GRng}(k)$. Consider distribution $D^1 = \{D^1_k\}_{k \in \mathbb{N}}$ such that D^1_k outputs (z, r^*) where r^* is chosen uniformly random from domain $\mathsf{GDom}(k)$ and $z = (f, K_H, f(r^*), h^*)$ for $f \leftarrow_{\$} \mathsf{Kg}(1^k)$, $K_H \leftarrow_{\$} \mathcal{K}_H(1^k)$, and $h^* = H(K_H, r^*)$. We note that D^1 is unpredictable since $\mathcal{F}$ is one-way and $\mathcal{H}$ is a hardcore function for $\mathcal{F}$. Now, consider adversary C attacking one-wayness extractor $\mathcal{G}$ in Figure 11. Then, we obtain that $\Pr[\, G_2 \Rightarrow 1 \,] - \Pr[\, G_3 \Rightarrow 1 \,] \leq \mathbf{Adv}^{\mathsf{cdist}}_{\mathcal{G},C,D^1}$.

Next, we give an adversary attacking SE to bound the probability of game G_3 outputs 1. Consider IND-CPA adversary D attacking SE in Figure 12. Then, we have $\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{SE},D}(k) = 2 \cdot \Pr[\, G_3 \Rightarrow 1 \,] - 1$. Summing up,

$$\mathbf{Adv}^{\text{ind-cca}}_{\overline{\mathsf{FO}},A}(k) \leq 2p \cdot \mathbf{Adv}^{\text{int-ctxt}}_{\mathsf{SE},B_1}(k) + 2 \cdot \mathbf{Adv}^{\text{cr}}_{\mathcal{H},B_2}(k) + 2 \cdot \mathbf{Adv}^{\text{ext2}}_{\mathcal{H},B,\mathsf{Ext}}(k) + 2 \cdot \mathbf{Adv}^{\mathsf{cdist}}_{\mathcal{G},C,D^1} + \mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{SE},D}(k) \ .$$

This completes the proof of Theorem 3.1. ▮

Adversary $C(K_G, z, K^*)$	**Procedure** $\mathsf{Dec}(c)$
$(f, K_H, f(r^*), h^*) \leftarrow z$; $\mathsf{coin} \leftarrow\$ \mathsf{Coins}(k)$	$(c_1, c_2) \leftarrow c$
$(pk', sk') \leftarrow\$ \mathsf{Kg}(1^k)$; $pk \leftarrow (pk', f, K_H, K_G)$	$h \leftarrow \mathsf{Rec}(sk', c_1)$
$\mathsf{hint} \leftarrow (f, K_G, f(r^*), K^*)$; $aux \leftarrow (b, pk', sk')$	$r \leftarrow \mathsf{Ext}(K_H, aux, \mathsf{coin}, \mathsf{hint}, h^*, h)$
$(\mathcal{M}_0, \mathcal{M}_1, st) \leftarrow A_1^{\mathsf{Dec}(\cdot)}(pk; \mathsf{coin})$	$K \leftarrow G(K_G, r)$; $m \leftarrow \mathcal{D}_K(c_2)$
$b \leftarrow\$ \{0,1\}$; $m_b \leftarrow \mathcal{M}_b(1^k, pk; \mathsf{coin})$	Return m
$c_1^* \leftarrow \mathsf{Enc}(pk', f(r^*); h^*)$; $c_2^* \leftarrow\$ \mathcal{E}_{K^*}(m_b)$	
$d \leftarrow A_2^{\mathsf{Dec}(\cdot)}(st, (c_1^*, c_2^*); \mathsf{coin})$	
Return $(b = d)$	

Figure 11: **Adversary C in the proof of Theorem 3.1.**

Adversary $D^{\mathcal{E}_{K^*}}(1^k)$	**Procedure** $\mathsf{Dec}(c)$
$K_H \leftarrow\$ \mathcal{K}_H(1^k)$; $K_G \leftarrow\$ \mathcal{K}_G(1^k)$	$(c_1, c_2) \leftarrow c$
$(f, f^{-1}) \leftarrow\$ \mathsf{Kg}(1^k)$; $(pk', sk') \leftarrow\$ \mathsf{Kg}(1^k)$	$(f(r), h) \leftarrow \mathsf{Dec}(sk', c_1)$
$pk \leftarrow (pk', f, K_H, K_G)$; $sk \leftarrow (sk', f^{-1}, K_H, K_G)$	$r \leftarrow f^{-1}(f(r))$
$(\mathcal{M}_0, \mathcal{M}_1, st) \leftarrow\$ A_1^{\mathsf{Dec}(\cdot)}(pk)$; $r^* \leftarrow\$ \mathsf{HDom}(k)$	$K \leftarrow G(K_G, r)$
$h^* \leftarrow H(K_H, r^*)$; $c_1^* \leftarrow \mathsf{Enc}(pk', f(r^*); h^*)$	$m \leftarrow \mathcal{D}_K(c_2)$
$c_2^* \leftarrow \mathcal{E}_{K^*}(\mathcal{M}_0, \mathcal{M}_1)$; $d \leftarrow\$ A_2^{\mathsf{Dec}(\cdot)}(st, (c_1^*, c_2^*))$	Return m
Return $(b = d)$	

Figure 12: **Adversary D in the proof of Theorem 3.1.**

4 RSA-OAEP Instantiation

In this section, we partially instantiate RSA-OAEP using extractable functions. Our result uses extractability assumption on $\mathcal{G}$ while modeling $\mathcal{H}$ as random oracle.

Theorem 4.1 *Let $n, \mu, \zeta, \rho, \eta$ be integer parameters. Let $\mathcal{G} : \mathcal{K}_G \times \{0,1\}^\rho \to \{0,1\}^{\mu+\zeta}$ be a hash function family 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 $\mathcal{G}$ is PRG, (η, ζ)-EXT2 and ζ-NCR. Moreover, suppose $\mathcal{F}$ is ζ-POW. Then $\mathsf{OAEP}[\mathcal{G}, \mathcal{H}, \mathcal{F}]$ is IND-CCA secure.*

Proof: We prove security through a sequence of games. Consider games G_1–G_5 in Figure 13. Each game maintains two independent random oracles RO and $\overline{\text{RO}}$. Procedure RO maintains a local array H as follows:

Procedure $\text{RO}(v)$
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: Game G_1 is the standard indistinguishability chosen ciphertext (IND-CCA) game. Thus, we have that $\mathbf{Adv}^{\text{ind-cca}}_{\mathsf{OAEP},A}(k) = 2 \cdot \Pr[G_1 \Rightarrow 1] - 1$ for any PPT adversary A.

Game G_2: Game G_2 is similar 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 [7],

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

Now consider adversary D_1 attacking pseudorandom generator $\mathcal{G}$ in Figure 14. We know that $\mathbf{Adv}^{\text{prg}}_{\mathcal{G},D_1}(k) = 2 \cdot \Pr\left[\text{PRG-DIST}^{D_1}_{\mathcal{G}}(k) \Rightarrow 1\right] - 1$. Let $\text{PRG-REAL}^{D_1}_{\mathcal{G}}$ be the game identical to game $\text{PRG-DIST}^{D_1}_{\mathcal{G}}$ condition on $b = 1$ and $\text{PRG-RAND}^{D_1}_{\mathcal{G}}$ be the game identical to game $\text{PRG-DIST}^{D_1}_{\mathcal{G}}$ condition on $b = 0$. Then,

$$\mathbf{Adv}^{\text{prg}}_{\mathcal{G},D_1}(k) = \Pr\left[\text{PRG-REAL}^{D_1}_{\mathcal{G}} \Rightarrow 1\right] - \Pr\left[\text{PRG-RAND}^{D_1}_{\mathcal{G}} \Rightarrow 1\right] \ .$$

Games $G_1(k)$, $G_2(k)$

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

Games $G_3(k), G_4(k)$, $G_5(k)$

$\mathsf{coin} \leftarrow\!\!\$\; \mathsf{Coins}$; $b \leftarrow\!\!\$\; \{0,1\}$; $K_G \leftarrow\!\!\$\; \mathcal{K}_G(1^k)$
$(f, f^{-1}) \leftarrow\!\!\$\; \mathsf{Kg}(1^k)$; $pk \leftarrow (K_G, f)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\; A_1^{\mathrm{RO}_1(\cdot),\mathsf{Dec}(\cdot)}(1^k, pk; \mathsf{coin})$
$m_b \leftarrow\!\!\$\; \mathcal{M}_b^{\mathrm{RO}_1(\cdot)}(1^k, pk; \mathsf{coin})$; $r^* \leftarrow\!\!\$\; \{0,1\}^\rho$
$x^* \leftarrow G(K_G, r^*)$; $x^* \leftarrow\!\!\$\; \{0,1\}^{\mu+\zeta}$
$s^* \leftarrow x^* \oplus (0^\zeta \| m_b)$; $t^* \leftarrow\!\!\$\; \{0,1\}^\rho$
$z^* \leftarrow t^* \oplus r^*$; $H[s^*] \leftarrow z^*$; $c^* \leftarrow f(s^* \| t^*)$
$d \leftarrow\!\!\$\; A_2^{\mathrm{RO}_2(\cdot),\mathsf{Dec}(\cdot)}(c^*, state; \mathsf{coin})$
Return $(b = d)$

Procedure $\mathrm{RO}_1(s)$ // of games G_1, G_2
Return $\mathrm{RO}(s)$

Procedure $\mathrm{RO}_2(s)$ // of games G_1, G_2
Return $\mathrm{RO}(s)$

Procedure $\mathrm{RO}_2(s)$ // of games G_3, G_4, G_5
$\mathbf{s} \leftarrow s \cup \mathbf{s}$; $\mathbf{z} \leftarrow \mathrm{RO}(s) \cup \mathbf{z}$
If $s = s^*$ then
  $\mathsf{bad}_2 \leftarrow \mathsf{true}$; return $\overline{\mathrm{RO}}(s)$
Return $\mathrm{RO}(s)$

Procedure $\mathsf{Dec}(c)$ // of games G_3–G_5
For all $s \in \mathbf{s}$ do
  $r \leftarrow \mathsf{Ext}(K_G, \mathbf{z}, b, f, t^*, \mathsf{coin}, x^*, s|^\zeta)$
  $m \leftarrow G(K_G, r)|_\mu \oplus s|_\mu$
  If $\mathsf{Enc}(pk, m; r) = c$ then return m
Return $\perp$

Procedure $\mathrm{RO}_1(s)$ // of games G_3–G_5
$\mathbf{s} \leftarrow s \cup \mathbf{s}$; $\mathbf{z} \leftarrow \mathrm{RO}(s) \cup \mathbf{z}$
Return $\mathrm{RO}(s)$

Figure 13: **Games G_1–G_5 in the proof of Theorem 4.1.**

Adversary $D_1(K_G, x^*)$
$(f, f^{-1}) \leftarrow\!\!\$\; \mathsf{Kg}(1^k)$; $\mathsf{out} \leftarrow 0$
$pk \leftarrow (K_G, f)$; $sk \leftarrow (K_G, f^{-1})$; $b \leftarrow\!\!\$\; \{0,1\}$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\; A_1^{\mathrm{ROSIM}_1(\cdot),\mathsf{Dec}(\cdot)}(1^k, pk)$
$m_b \leftarrow\!\!\$\; \mathcal{M}_b^{\mathrm{ROSIM}_1(\cdot)}(1^k, pk)$
$s^* \leftarrow x^* \oplus (0^\zeta \| m_b)$
If $H[s^*] \neq \perp$ then $\mathsf{out} \leftarrow 1$
Return out

Procedure $\mathsf{Dec}(c)$
$m \leftarrow \mathsf{Dec}(sk, c)$
Return m

Procedure $\mathrm{ROSIM}_1(s)$
If $H[s] = \perp$ then
  $H[s] \leftarrow\!\!\$\; \{0,1\}^\rho$
Return $H[s]$

Figure 14: **Adversary D_1 in the proof of Theorem 4.1.**

Note that $\Pr\left[\text{PRG-REAL}_{\mathcal{G}}^{D_1} \Rightarrow 1\right] = \Pr[\,G_2(k) \text{ sets } \mathsf{bad}_1\,]$. Moreover, observe that in the $\text{PRG-RAND}_{\mathcal{G}}^{D_1}$, the probability adversary A queries for s^* is uniformly random. Multiplying for q random-oracle queries we have $\Pr[\text{PRG-RAND}_{\mathcal{G}}^{D_1} \Rightarrow 1] \leq q/2^{\mu+\zeta}$. Thus, we have $\Pr[\,G_2(k) \text{ sets } \mathsf{bad}_1\,] \leq \mathbf{Adv}_{\mathcal{G},D_1}^{\mathrm{prg}}(k) + q/2^{\mu+\zeta}$.

Game G_3: Game G_3 is similar to game G_2 except that we made two changes. First, we reorder the code of game G_2 in producing t^*. The change is conservative and won't effect probability of game G_3 outputting 1 compare to game G_2. Second, we change the decryption oracle as follows. Let $\mathbf{s}$ be the array of random oracle queries made by adversary A. For each RO query $s \in \mathbf{s}$, we run the extractor for hash function family $\mathcal{G}$ on ζ-most significant bits of s to extract randomness r and then compute message m. Consider EXT2 adversary B in Figure 15. Let Ext be an extractor for adversary B. We note that adversary B gets no hints from image oracle $\mathcal{I}$. We define F to be the event where algorithm Dec fails to successfully decrypt on at least one challenge ciphertext. Then, we have $\Pr[\,G_2(k) \Rightarrow 1\,] - \Pr[\,G_3(k) \Rightarrow 1\,] \leq \Pr[\,F\,]$.

Let c_i be the i-th decryption query that adversary A makes, r_i be the corresponding randomness and s_i be the $\mu{+}\zeta$-most significant bits of $f^{-1}(c_i)$. For all $i \in [q]$ we define $E_{1,i}$ to be the event where $s_i \notin \mathbf{s}$. Moreover, we define E_2 to be the event where there exists at least one decryption query c_i such that $s_i = s^*$. Observe that since $c_i \neq c^*$ it implies that $r_i \neq r^*$. Therefore if E_2 happens then there is the NCR adversary C that

Adversary $B^{\mathcal{O},\mathcal{I}}(K_G, aux; \mathsf{coin})$
$i \leftarrow 0$; $\mathbf{s} \leftarrow \perp$; $(\mathbf{z}, b, f, t^*) \leftarrow aux$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\$ A_1^{\text{ROSim}(\cdot),\mathsf{Dec}(\cdot)}(1^k, pk; \mathsf{coin})$
$m_b \leftarrow\$ \mathcal{M}_b^{\text{ROSim}(\cdot)}(1^k, pk; \mathsf{coin})$
$x^* \leftarrow\$ \mathcal{I}(1^k)$; $s^* \leftarrow x^* \oplus (0^\zeta \| m_b)$; $c^* \leftarrow f(s^* \| t^*)$
Run $A_2^{\text{ROSim}(\cdot),\mathsf{Dec}(\cdot)}(c^*, state; \mathsf{coin})$
 	
Procedure $\text{ROSim}(s)$
If $s = s^*$ then Halt
$i \leftarrow i+1$; $\mathbf{s} \leftarrow s \cup \mathbf{s}$
Return $\mathbf{z}[i]$
	Procedure $\mathsf{Dec}(c)$
	For all $s \in \mathbf{s}$ do
	   $r \leftarrow \mathcal{O}(s\|^\zeta)$; $m \leftarrow G(K_G, r)\|_\mu \oplus s\|_\mu$
	   If $\mathsf{Enc}(pk, m; r) = c$ then return m
	Return $\perp$

Figure 15: **Adversary B in the proof of Theorem 4.1.**

Adversary $D_2(K_G, x^*)$
For $i \in [q]$ do $\mathbf{z}[i] \leftarrow\$ \{0,1\}^\rho$
$(f, f^{-1}) \leftarrow\$ \mathsf{Kg}(1^k)$; $\mathsf{out} \leftarrow 0$; $i \leftarrow 0$
$pk \leftarrow (K_G, f)$; $b \leftarrow\$ \{0,1\}$; $t^* \leftarrow\$ \{0,1\}^\rho$
$\mathsf{coin} \leftarrow\$ \mathsf{Coins}$; $aux \leftarrow (\mathbf{z}, b, f, t^*)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\$ A_1^{\text{ROSim}_1(\cdot),\mathsf{Dec}(\cdot)}(1^k, pk; \mathsf{coin})$
$m_b \leftarrow\$ \mathcal{M}_b^{\text{ROSim}_1(\cdot)}(1^k, pk)$
$s^* \leftarrow x^* \oplus (0^\zeta \| m_b)$; $c^* \leftarrow f(s^* \| t^*)$
Run $A_2^{\text{ROSim}_2(\cdot),\mathsf{Dec}(\cdot)}(c^*, state; \mathsf{coin})$
Return out
 	
Procedure $\text{ROSim}_1(s)$
If $H[s] = \perp$ then
   $i \leftarrow i+1$; $\mathbf{s}[i] \leftarrow s$; $H[s] \leftarrow \mathbf{z}[i]$
Return $H[s]$
	Procedure $\text{ROSim}_2(s)$
	If $s = s^*$ then
	   $\mathsf{out} \leftarrow 1$; Halt run of A_2
	If $H[s] = \perp$ then
	   $i \leftarrow i+1$; $\mathbf{s}[i] \leftarrow s$; $H[s] \leftarrow \mathbf{z}[i]$
	Return $H[s]$
	
	Procedure $\mathsf{Dec}(c)$
	For all $s \in \mathbf{s}$ do
	   $r \leftarrow \mathsf{Ext}(K_G, aux, \mathsf{coin}, x^*, s\|^\zeta)$
	   $m \leftarrow G(K_G, r)\|_\mu \oplus s\|_\mu$
	   If $\mathsf{Enc}(pk, m; r) = c$ then return m
	Return $\perp$

Figure 16: **Adversary D_2 in the proof of Theorem 4.1.**

finds collision. Thus, we have $\Pr[E_2] \le \mathbf{Adv}^{\text{n-cr}}_{\mathcal{G},C}(k)$. On the other hand, when $E_{1,i}$ happens algorithm Dec outputs $\perp$. Let $E_1 = \cup^{i=1} E_{1,i}$. We have from [18, Theorem 3.4] that when E_1 and $\overline{E_2}$ happens the ciphertext c_i is a valid ciphertext at most with probability $1/2^\zeta$. Then we have

$$\Pr[F \wedge E_1] \le \Pr[F \wedge E_1 \wedge E_2] + \Pr\left[F \wedge E_1 \wedge \overline{E_2}\right] \le \mathbf{Adv}^{\text{n-cr}}_{\mathcal{G},C}(k) + q/2^\zeta \ .$$

We now define E_3 to be the event where there exists at least one decryption query c_i such that $s_i|^\zeta = s^*|^\zeta$. Observe that if $\overline{E_1}$ and E_3 happens then there is the POW adversary I_1 attacking the TDP. Thus, we have $\Pr\left[\overline{E_1} \wedge E_3\right] \le q \cdot \mathbf{Adv}^{\text{pow}}_{\mathcal{F},I_1}(k)$. Note that for all decryption query c_i where $s_i \in \mathbf{s}$ and $s_i|^\zeta \ne s^*|^\zeta$, the extractor Ext can successfully extract r_i with high probability. Then we have

$$\Pr\left[F \wedge \overline{E_1}\right] \le \Pr\left[\overline{E_1} \wedge E_3\right] + \Pr\left[F \wedge \overline{E_1} \wedge \overline{E_3}\right] \le q \cdot \mathbf{Adv}^{\text{pow}}_{\mathcal{F},I_1}(k) + \mathbf{Adv}^{\text{ext2}}_{\mathcal{G},B,\mathsf{Ext}}(k) \ .$$

Game G_4**:** Game G_4 is similar to game G_3 except in procedure RO_2, 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, and based on Fundamental Lemma of Game-playing [7], we have $\Pr[G_3(k) \Rightarrow 1] - \Pr[G_4(k) \Rightarrow 1] \le \Pr[G_4(k) \text{ sets } \mathsf{bad}_2]$. Consider adversary D_2 attacking the pseudorandom generator $\mathcal{G}$ in Figure 16. Let $\text{PRG-REAL}^{D_2}_{\mathcal{G}}$ be the game identical to game $\text{PRG-DIST}^{D_2}_{\mathcal{G}}$ condition on $b = 1$, and $\text{PRG-RAND}^{D_2}_{\mathcal{G}}$ be the game identical to game $\text{PRG-DIST}^{D_2}_{\mathcal{G}}$ condition on $b = 0$. Then, $\mathbf{Adv}^{\text{prg}}_{\mathcal{G},D_2}(k) = \Pr\left[\text{PRG-REAL}^{D_2}_{\mathcal{G}} \Rightarrow 1\right] - \Pr\left[\text{PRG-RAND}^{D_2}_{\mathcal{G}} \Rightarrow 1\right]$. Note that $\Pr\left[\text{PRG-REAL}^{D_2}_{\mathcal{G}} \Rightarrow 1\right] = \Pr[G_4(k) \text{ sets } \mathsf{bad}_2]$.

To bound the probability of game $\text{PRG-RAND}^{D_2}_{\mathcal{G}}$ outputs 1, we construct inverter I_2 attacking the family

Inverter $I_2(f, c^*)$
For $i \in [q]$ do $\mathbf{z}[i] \leftarrow\!\!\$\; \{0,1\}^\rho$
$b \leftarrow\!\!\$\; \{0,1\}$; $i \leftarrow 0$; $\mathsf{out} \leftarrow \perp$; $j \leftarrow\!\!\$\; [q]$
$K_G \leftarrow\!\!\$\; \mathcal{K}_G(1^k)$; $pk \leftarrow (K_G, f)$
$\mathsf{coin} \leftarrow\!\!\$\; \mathsf{Coins}$; $aux \leftarrow (\mathbf{z}, b, f, c^*)$
$(\mathcal{M}_0, \mathcal{M}_1, state) \leftarrow\!\!\$\; A_1^{\mathrm{ROSIM}_1(\cdot), \mathsf{Dec}(\cdot)}(1^k, pk; \mathsf{coin})$
$m_b \leftarrow\!\!\$\; \mathcal{M}_b^{\mathrm{ROSIM}_1(\cdot)}(1^k, pk; \mathsf{coin})$
Run $A_2^{\mathrm{ROSIM}_2(\cdot), \mathsf{Dec}(\cdot)}(c^*, state; \mathsf{coin})$
Return out
 	
Procedure $\mathrm{ROSIM}_1(s)$
If $H[s] = \perp$ then
    $i \leftarrow i+1$; $\mathbf{s}[i] \leftarrow s$; $H[s] \leftarrow \mathbf{z}[i]$
Return $H[s]$
	Procedure $\mathrm{ROSIM}_2(s)$
	If $H[s] = \perp$ then
	    $i \leftarrow i+1$; $\mathbf{s}[i] \leftarrow s$; $H[s] \leftarrow \mathbf{z}[i]$
	If $i = j$ then
	    $\mathsf{out} \leftarrow s\|^\zeta$; Halt run of A_2
	Return $H[s]$
	
	Procedure $\mathsf{Dec}(c)$
	For all $s \in \mathbf{s}$ do
	    $r \leftarrow \overline{\mathsf{Ext}}(K_G, aux, \mathsf{coin}, s\|^\zeta)$
	    $m \leftarrow G(K_G, r)\|_\mu \oplus s\|_\mu$
	    If $\mathsf{Enc}(pk, m; r) = c$ then return m
	Return $\perp$

Figure 17: **Inverter I_2 in the proof of Theorem 4.1.**

of partial one-way trapdoor permutation $\mathcal{F}$ in Figure 17. We note that in the game $\mathrm{PRG\text{-}RAND}_{\mathcal{G}}^{D_2}$, the challenge c^* is independent of K_G thus in the decryption oracle it is suffices to use the EXT1 extractor where c^* is an auxiliary information. The EXT1 adversary and extractor are similar to the one given in game G_3. Observe that if adversary A_2 queries for s^* then inverter I could partially invert challenge c^*. Hence, $\Pr\left[\mathrm{PRG\text{-}RAND}_{\mathcal{G}}^{D_2} \Rightarrow 1\right] \le q \cdot \mathbf{Adv}^{\mathrm{pow}}_{\mathcal{F}, I_2}(k)$. Thus,

$$\Pr\left[\, G_4(k) \text{ sets } \mathsf{bad}_2 \,\right] \le \mathbf{Adv}^{\mathrm{prg}}_{\mathcal{G}, D_2}(k) + q \cdot \mathbf{Adv}^{\mathrm{pow}}_{\mathcal{F}, I_2}(k) \ .$$

Game G_5: Game G_5 is similar to game G_4 except except we are using completely random x^* in the encryption phase instead of using the pseudorandom value $G(K_G, r^*)$. Consider adversary D_3 attacking the pseudorandom generator $\mathcal{G}$ similar to D_2. Then

$$\Pr[G_4(k) \Rightarrow 1] - \Pr[G_5(k) \Rightarrow 1] \le \mathbf{Adv}^{\mathrm{prg}}_{\mathcal{G}, D_3}(k) \ .$$

Note that $\Pr[G_5(k) \Rightarrow 1] = 1/2$, since the distribution of ciphertexts is completely independent of bit b. Summing up,

$$\mathbf{Adv}^{\mathrm{ind\text{-}cca}}_{\mathsf{OAEP}, A}(k) \le 4q \cdot \mathbf{Adv}^{\mathrm{pow}}_{\mathcal{F}, I}(k) + 6 \cdot \mathbf{Adv}^{\mathrm{prg}}_{G, D}(k) + 2 \cdot \mathbf{Adv}^{\mathrm{n\text{-}cr}}_{\mathcal{G}, C}(k) + 2 \cdot \mathbf{Adv}^{\mathrm{ext2}}_{\mathcal{G}, B, \mathsf{Ext}}(k) + \frac{2q}{2^{\mu+\zeta}} + \frac{2q}{2^\zeta} \ .$$

This completes the proof. ▮

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 4.)

[2] 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 3.)

[3] 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.)

[4] 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 4.)

[5] 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.)

[6] 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, 8.)

[7] 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 11, 13.)

[8] 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 3.)

[9] 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 4.)

[10] 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.)

[11] C. Brzuska, P. Farshim, and A. Mittelbach. Random-oracle uninstantiability from indistinguishability obfuscation. In *Theory of Cryptography - 12th Theory of Cryptography Conference, TCC*, volume 9015 of *Lecture Notes in Computer Science*, pages 428–455. Springer, 2015. (Cited on page 3.)

[12] 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 4.)

[13] 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.)

[14] 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 3.)

[15] 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 3.)

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

[17] 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 4.)

[18] N. Cao, A. O'Neill, and M. Zaheri. Toward rsa-oaep without random oracles. Cryptology ePrint Archive, Report 2018/1170, 2018. `https://eprint.iacr.org/2018/1170`. (Cited on page 3, 4, 8, 13.)

[19] D. Dachman-Soled, G. Fuchsbauer, P. Mohassel, and A. O'Neill. Enhanced chosen-ciphertext security and applications. In H. Krawczyk, editor, *PKC 2014*, volume 8383 of *LNCS*, pages 329–344, Buenos Aires, Argentina, Mar. 26–28, 2014. Springer, Heidelberg, Germany. (Cited on page 6.)

[20] I. Damgård, S. Faust, and C. Hazay. Secure two-party computation with low communication. In R. Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 54–74, Taormina, Sicily, Italy, Mar. 19–21, 2012. Springer, Heidelberg, Germany. (Cited on page 3.)

[21] 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 4.)

[22] E. Fujisaki and T. Okamoto. Secure integration of asymmetric and symmetric encryption schemes. *Journal of Cryptology*, 26(1):80–101, Jan. 2013. (Cited on page 3, 9.)

[23] 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 6.)

[24] 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 7.)

[25] S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984. (Cited on page 6.)

[26] 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 4.)

[27] D. Hofheinz, K. Hövelmanns, and E. Kiltz. A modular analysis of the fujisaki-okamoto transformation. In *TCC (1)*, pages 341–371. Springer, 2017. (Cited on page 3.)

[28] C.-Y. Hsiao, C.-J. Lu, and L. Reyzin. Conditional computational entropy, or toward separating pseudoentropy from compressibility. In M. Naor, editor, *EUROCRYPT 2007*, volume 4515 of *LNCS*, pages 169–186, Barcelona, Spain, May 20–24, 2007. Springer, Heidelberg, Germany. (Cited on page 7.)

[29] 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 3, 8.)

[30] 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 6.)

[31] 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 3.)