Reset Indifferentiability from Weakened Random Oracle Salvages One-pass Hash Functions

Yusuke Naito, Kazuki Yoneyama, and Kazuo Ohta

Mitsubishi Electric Corporation, NTT Corporation and UEC

Abstract. Ristenpart et al. showed that the limitation of the indifferentiability theorem of Maurer et al. which does not cover all multi-stage security notions $\mathbf{S}_m$ but covers only single-stage security notions $\mathbf{S}_s$, defined reset indifferentiability, and proved the reset indifferentiability theorem, which is an analogy of the indifferentiability theorem covers all security notions $\mathbf{S}$ $(= \mathbf{S}_s \cup \mathbf{S}_m)$: $F_1 \sqsubset_r F_2 \Rightarrow \forall \mathcal{C} \in \mathbf{C}, \forall \mathcal{S} \in \mathbf{S}$: $\mathcal{C}(F_1) \succ_{\mathcal{S}} \mathcal{C}(F_2)$ (if a hash function $H^{\mathcal{U}}$ is reset indifferentiable from a random oracle $\mathcal{RO}$, $\mathcal{C} \in \mathbf{C}$ which is a set of all cryptosystems is at least as $\mathcal{S}$-secure in the $\mathcal{U}$ model as in the $\mathcal{RO}$ model). Unfortunately, they also proved the impossibility of $H^{\mathcal{U}} \sqsubset_r \mathcal{RO}$ where H is a one-pass hash construction such as ChopMD and Sponge.

In this paper, we will propose a new proof of molular approach instead of the $\mathcal{RO}$ methodology, "Reset Indifferentiability from Weakened Random Oracle", called as the $\mathcal{WRO}$ methodology, in order to ensure the $\mathcal{S}$-security of $\mathcal{C}$ with $H^{\mathcal{U}}$, salvaging ChopMD and Sponge. The concrete proof procedure of the $\mathcal{WRO}$ methodology is as follows:

1. Define a new concept of $\mathcal{WRO}$ instead of $\mathcal{RO}$,
2. Prove that $H^{\mathcal{U}} \sqsubset_r \mathcal{WRO}$, (here an example of H is ChopMD and Sponge), and
3. Prove that $\mathcal{C}$ is $\mathcal{S}$-secure in the $\mathcal{WRO}$ model.

As a result we can prove that $\mathcal{C}$ with $H^{\mathcal{U}}$ is $\mathcal{S}$-secure by combining the results of Steps 2, 3, and the theorem of Ristenpart et al. Moreover, for public-key encryption (as cryptosystem $\mathcal{C}$) and chosen-distribution attack (as the game of $\mathcal{S} \in \mathbf{S}_m$) we will prove that $\mathcal{C}(\mathcal{WRO})$ is $\mathcal{S}$-secure, which implies the appropriateness of the new concept of the $\mathcal{WRO}$ model.

Keywords. Indifferentiable hash function, reset indifferentiability, multi-stage game, Sponge, ChopMD.

1 Introduction

1.1 Indifferentiability

The Indifferentiability theorem [17] of Maurer, Renner, and Holenstein (MRH), called MRH theorem, ensures that for all single-stage security notions $\mathbf{S}_s$ the security of a cryptosystem in the F_2 model is ensured in the F_1 model by proving that "F_1 is indifferentiable from F_2", denoted by $F_1 \sqsubset F_2$. In the framework with which the MRH theorem is proven, two interfaces of a system F among adversaries and honest parities are considered, an "adversarial interface" accessed by adversaries, denoted by $F.adv$, and an "honest interface" accessed by honest parties, denoted by $F.hon$. The indifferentiable game of $F_1 \sqsubset F_2$ is a simulation based game where constructing some stateful simulator S which represents some adversary in the F_2 model which can obtain any information on $F_1.adv$ in the F_1 model by using information on $F_2.adv$. The definition of $F_1 \sqsubset F_2$ is that there exists a stateful simulator S such that for any distinguisher $\mathcal{D}$ which interacts with two oracles (L, R), no $\mathcal{D}$ can distinguish a real world $(L, R) = (F_1.hon, F_2.adv)$ from an ideal world $(L, R) = (F_2.hon, S^{F_2.adv})$ where S has access to $F_2.adv$. MRH proved the following theorem:

MRH Theorem.[17] $F_1 \sqsubset F_2 \Rightarrow \forall \mathcal{C} \in \mathbf{C}, \forall \mathcal{S} \in \mathbf{S}_s$: $\mathcal{C}(F_1) \succ_{\mathcal{S}} \mathcal{C}(F_2)$

which means $\mathcal{C}$ is at least as $\mathcal{S}$-secure in F_1 model as in F_2 model, where $\mathbf{C}$ is a set of all cryptosystems.

1.2 $\mathcal{RO}$ Methodology

Coron, Dodis, Malinaud, and Puniya [12] pointed out that the MRH theorem opened a nice molular approach of a security proof of some cryptosystem using hash fuctions, that is, since the MRH theorem claims that $H^{\mathcal{U}} \sqsubset \mathcal{RO} \Rightarrow \forall \mathcal{C} \in \mathbf{C}, \forall \mathcal{S} \in \mathbf{S}_s$: $\mathcal{C}(H^{\mathcal{U}}) \succ_{\mathcal{S}} \mathcal{C}(\mathcal{RO})$ assuming $\mathcal{U}$ is ideal, designers of hash functions concentrate on constructions of H proving $H^{\mathcal{U}} \sqsubset \mathcal{RO}$, and those of cryptosystems concentrate on finding of $\mathcal{C}$ proving that $\mathcal{C}$ is $\mathcal{S}$-secure in the $\mathcal{RO}$ model. This approach is called as the Random Oracle ($\mathcal{RO}$) methodology. In the proof of $H^{\mathcal{U}} \sqsubset \mathcal{RO}$, the real world is $(L, R) = (H^{\mathcal{U}}, \mathcal{U})$ and the ideal world is $(L, R) = (\mathcal{RO}, S^{\mathcal{RO}})$. Hereafter, we call a hash function $H^{\mathcal{U}}$ such that $H^{\mathcal{U}} \sqsubset \mathcal{RO}$ an "IFRO (indifferentiable from a $\mathcal{RO}$) hash function" and its construction the "IFRO hash construction". So far, many IFRO hash constructions have been proposed such as the Chop Merkle-Damgård (ChopMD) construction [12] and the Sponge construction [6]. SHA-512/224 and SHA-512/256, which are standarized in FIPS 180-4 [20], employ the ChopMD construction and the SHA-3 winner Keccak [19, 7] employs the Sponge construction. Therefore, IFRO security is an important criterion of design of hash functions.

1.3 Impossibility of IFRO security in Multi-Stage Security Games

However, Ristenpart, Shacham, and Shrimpton (RSS) [21] pointed out that though indifferentiability covers all single-stage security notions $\mathbf{S}_s$, it does not cover all multi-stage security notions $\mathbf{S}_m$. In a multi-stage game, the size of the state shared among adversaries is restricted.

The reason why indifferentiability does not cover $\mathbf{S}_m$ is that the indifferentiable game deal with a "stateful" simulator, that is the size of the sate of the simulator is not restricted, while in the multi-stage game the size of the state shared among adversaries is restricted.

They defined a two party challenge response protocol $\mathcal{CR}$ and its security, called CRP-security as a counter example against indifferentiability. They showed that $\mathcal{CR}$ is CRP-secure in the $\mathcal{RO}$ model but insecure when using an IFRO hash function such as the ChopMD hash function and the Sponge hash function.

Note that the RSS result does not always imply that for any $\mathcal{S} \in \mathbf{S}_m$, any $\mathcal{C} \in \mathbf{C}$ which is $\mathcal{S}$-secure in the $\mathcal{RO}$ model is insecure when $\mathcal{RO}$ is replaced by $H^{\mathcal{U}}$. So we have the following question:

"Can we prove the $\mathcal{S}$-security of $\mathcal{C}$ with $H^{\mathcal{U}}$?"

This paper tackles how to solve this question. The candidate to solve this question is reset indifferentiability of RSS [21].

1.4 Reset Indifferentiability

The reset indifferentiability framework is an extension of the indifferentiability framework and this theorem, called RSS theorem, covers all security notions $\mathbf{S}$ $(= \mathbf{S}_s \cup \mathbf{S}_m)$. The RSS theorem ensures that for any $\mathcal{S} \in \mathbf{S}$ and for any $\mathcal{C}$, the $\mathcal{S}$-security is preserved when F_2 is replaced with F_1 if F_1 is reset indifferentiable from F_2, denoted by $F_1 \sqsubset_r F_2$. The reset indifferentiable game is the same simulation based game as the indifferentiable game [17], where indifferentiability deals with a stateful simulator, while reset indifferentiability deals with a *stateless* simulator. The "stateless" setting reflects the setting of multi-stage security games where the state size among adversaries is restricted. So the definition of $F_1 \sqsubset_r F_2$ is that there exists a stateless simulator S such that for any distinguisher $\mathcal{D}$ which interacts with two oracles (L, R), no $\mathcal{D}$ can distinguish a real world $(L, R) = (F_1.hon, F_2.adv)$ from an ideal world $(L, R) = (F_2.hon, S^{F_2.adv})$. RSS proved the following theorem.

RSS Theorem. [21] $F_1 \sqsubset_r F_2 \Rightarrow \forall \mathcal{C} \in \mathbf{C}, \forall \mathcal{S} \in \mathbf{S}$: $\mathcal{C}(F_1) \succ_{\mathcal{S}} \mathcal{C}(F_2)$.

Therefore, if $H^{\mathcal{U}} \sqsubset_r \mathcal{RO}$, for any $\mathcal{S} \in \mathbf{S}$ and any $\mathcal{C} \in \mathbf{C}$ the $\mathcal{S}$-security of $\mathcal{C}$ is preserved when $\mathcal{RO}$ is replaced with $H^{\mathcal{U}}$.

Unfortunately, RSS also proved the *impossibility* of $H^{\mathcal{U}} \sqsubset_r \mathcal{RO}$ where H is a one-pass hash construction such as the ChopMD construction and the Sponge construction. That is, it is impossible to simulate information of $\mathcal{U}$ $(= H^{\mathcal{U}}.adv)$ from that of $\mathcal{RO}$ $(= \mathcal{RO}.adv)$. Therefore, we have to consider another solution than the $\mathcal{RO}$ methodology.

1.5 Our Contributions – A New Proposal of $\mathcal{WRO}$ Methodology –

We propose a $\mathcal{WRO}$ methodology which is based on "Reset Indifferentiability from Weakened Random Oracle ($\mathcal{WRO}$)" in order to ensure the $\mathcal{S}$-security of $\mathcal{C}$ with $H^{\mathcal{U}}$. This paper deals with the ChopMD construction and the fixed output length Sponge (FOLSponge) constructions as $H^{\mathcal{U}}$, because these are employed in important hash functions such as SHA-512/224, SHA-512/256, which are in FIPS 180-4, and SHA-3 winner Keccak.

The concrete proof procedure of the $\mathcal{WRO}$ methodology is as follows:

1. Define a new concept of $\mathcal{WRO}$ instead of $\mathcal{RO}$,
2. Prove that $H^{\mathcal{U}} \sqsubset_r \mathcal{WRO}$ assuming $\mathcal{U}$ is ideal, and
3. Prove that $\mathcal{C}$ is $\mathcal{S}$-secure in the $\mathcal{WRO}$ model.

As a result we can prove that $\mathcal{C}$ with $H^{\mathcal{U}}$ is $\mathcal{S}$-secure by combining the results of Steps 2, 3, and the RSS theorem. Moreover, for public-key encryption (as cryptosystem $\mathcal{C}$) and Chosen Distribution Attack [1, 2] (as game $\mathcal{S}$) we will prove that $\mathcal{C}(\mathcal{WRO})$ is $\mathcal{S}$-secure, which implies the appropriateness of the new concept of the $\mathcal{WRO}$ model.

We define $\mathcal{WRO}$ so that one can construct a stateless simulator such that $H^{\mathcal{U}} \sqsubset_r \mathcal{WRO}$, that is, an adversary can simulate information of $\mathcal{U}$ $(= H^{\mathcal{U}}.adv)$ from $\mathcal{WRO}.adv$. We define $\mathcal{WRO}$ which consists of $\mathcal{RO}$ and sub oracle $\mathcal{O}^*$ which leaks information to simulate $\mathcal{U}$. The interfaces are defined as $\mathcal{WRO}.hon = \mathcal{RO}$ and $\mathcal{WRO}.adv = (\mathcal{RO}, \mathcal{O}^*)$. If we can construct such $\mathcal{WRO}$, for any $\mathcal{S} \in \mathbf{S}$ and any $\mathcal{C} \in \mathbf{C}$, the $\mathcal{S}$-security is preserved when $\mathcal{WRO}$ is replaced with $H^{\mathcal{U}}$ by the RSS theorem.

To our knowledge, our result is the first result to ensure the reducibility from a real model to an ideal model for the important hash constructions, ChopMD and FOLSponge.

How to Define $\mathcal{O}^*$. We define $\mathcal{O}^*$ based on the IFRO proof of the ChopMD, ChopMD$^h \sqsubset \mathcal{RO}$, where $h : \{0,1\}^{m+2n} \to \{0,1\}^{2n}$ is a random oracle compression function. The output of the two block message $M_1\|M_2$ is calculated as ChopMD$^h(M_1\|M_2) = chop_n(h(h(IV\|M_1)\|M_2))$ where $chop_n$ accepts $2n$ bit value $x'\|x^*$ and returns the right n bit value x^*. In this case, the real world is $(L, R) = (\text{ChopMD}^h, h)$. In the indifferentiable game, since distinguisher $\mathcal{D}$ interacts with (L, R), helpful information for $\mathcal{D}$ is just query-response values from L and R. Therefore, the following two points are required to construct a simulator S. The first point is the simulation of h. The second point is the simulation of the relation between L and R in the real world, because L uses R in the real world. The following explains the simulations as considering the use of the S's state.

Simulation of h: We explain the simulation of h by using Fig. 1. This example is that $\mathcal{D}$ makes a repeated query. In the real world the responses y_1 and y_2 satisfy the following conditions, since R is a random oracle h,

- **Condition 1:** y_1 is a random value and $y_2 = y_1$.

The following demonstrates that S satisfying the condition can be constructed by using the S's state.

$\mathcal{D}$'s Procedure 1 (Condition 1)
1. $\mathcal{D}$ makes a query x to R and receives the response y_1.
2. $\mathcal{D}$ makes a query x to R and receives the response y_2.

Fig. 1. Distinguisher's Procedure 1

$\mathcal{D}$'s Procedure 2 (Condition 2)
1. $\mathcal{D}$ makes a query $IV \| M_1$ to R and receives the response y_1
2. $\mathcal{D}$ makes a query $y_1 \| M_2$ to R and receives the response y_2

Fig. 2. Distinguisher's Procedure 2

- **Constructing S:** In Step 1 S chooses a random value as the response y_1 for the query x. Then S *records* the query response pair (x, y_1). In Step 2 S finds y_1 from the query response pair (x, y_1) for the repeated query x, $y_2 := y_1$, and responds y_2.

Simulation of the L-R Relation: We explain the simulation of the relation between L and R by using Fig. 2. In the real world, since $(L, R) = (\mathrm{ChopMD}^h, h)$, the query response values in Fig. 2 satisfy the following conditions.

- **Condition 2:** $chop_n(y_1) = \mathrm{ChopMD}^f(M_1)$ and $chop_n(y_2) = \mathrm{ChopMD}^h(M_1 \| M_2)$.

The following shows that S satisfying the condition can be constructed by using the S's state.

- **Constructing S:** In Step 1 S defines $y_1^* = \mathcal{RO}(M_1)$ for the query $IV \| M_1$, chooses a random value y_1', defines $y_1 := y_1' \| y_1^*$, and returns y_1. Then S *records* the pair (M_1, y_1). In Step 2, for the query $y_1 \| M_2$, S finds M_1 from the pair (M_1, y_1). Then S chooses a random value y_2', defines $y_2^* = \mathcal{RO}(M_1 \| M_2)$, $y_2 := y_2' \| y_2^*$, and returns y_2. This procedure ensures that $chop_n(y_1) = \mathcal{RO}(M_1)$ and $chop_n(y_2) = \mathcal{RO}(M_1 \| M_2)$.

The important point of this simulation is to find M_1 from both the query $y_1 \| M_2$ and the recoded pair (M_1, y_1).

We can construct a stateful simulator S which ensures the two points. On the other hand, no one can construct a stateless simulator S which ensures the two points. So we compensate the stateless setting by using sub oracle $\mathcal{O}^*$.

Sub Oracle for Simulation of h: In order to ensure the condition 1, we add random oracle $\mathcal{RO}^*$ to $\mathcal{O}^*$. Then we can construct a stateless simulator S which ensures the condition 1; In Step 1 S defines $y_1 = \mathcal{RO}^*(x)$ for query x. In Step 2 S defines $y_2 = \mathcal{RO}^*(x)$ for the repeated query x.

Sub Oracle for Simulation of L-R Relation: In order to ensure the condition 2, we add random oracle $\mathcal{RO}^\dagger$ and trace oracle $\mathcal{TO}$ to $\mathcal{O}^*$. The definition of $\mathcal{TO}$ is that for query y_1' to $\mathcal{TO}$, $\mathcal{TO}$ returns M_1 if a query M_1 to $\mathcal{RO}^\dagger$ was made such that $y_1' = \mathcal{RO}^\dagger(M_1)$, otherwise $\mathcal{TO}$ returns $\perp$. Then we can construct a stateless simulator S which ensures the condition 2; In Step 1, for query $IV \| M_1$, S defines $y_1' = \mathcal{RO}^\dagger(M_1)$ and $y_1^* = \mathcal{RO}(M_1)$, $y_1 := y_1' \| y_1^*$, and returns y_1. In Step 2, for query $y_1 \| M_2$, S obtains y_1' from $y_1 := y_1' \| y_1^*$ and makes a query y_1' to $\mathcal{TO}$. Then M_1 is returned from $\mathcal{TO}$. Finally S defines $y_2^* = \mathcal{RO}(M_1 \| M_2)$ and $y_2' := \mathcal{RO}^\dagger(M_1 \| M_2)$, $y_2 := y_2' \| y_2^*$, and returns y_2. This procedure ensures that $chop_n(y_1) = \mathcal{RO}(M_1)$ and $chop_n(y_2) = \mathcal{RO}(M_1 \| M_2)$.

Therefore, we define $\mathcal{O}^* := (\mathcal{RO}^*, \mathcal{RO}^\dagger, \mathcal{TO})$, can construct a stateless simulator which ensures the above two conditions, and can succeed in proving of $\mathrm{ChopMD}^h \sqsubset_r \mathcal{WRO}$ (Theorem 2).

Similarly, by defining $\mathcal{O}^* := (\mathsf{IC}, \mathcal{RO}^\dagger, \mathcal{TO})$, for the FOLSponge construction, we can construct a stateless simulator which ensures the above two simulations and can succeed in proving of FOLSponge $\sqsubset_r \mathcal{WRO}$ (Theorem 3) where $\mathsf{IC} = (E, D)$ is an ideal cipher. E is an encryption oracle and D is a decryption oracle.

In this paper we define $\mathcal{O}^* = (\mathcal{RO}^*, \mathcal{RO}^\dagger, \mathcal{TO}, \mathsf{IC})$ in order to evaluate the ChopMD and the FOLSponge constructions by the single $\mathcal{WRO}$.[1] Since we defined $\mathcal{WRO} = (\mathcal{RO}, \mathcal{O}^*)$, $\mathcal{WRO}$ con-

[1] Similarly, one can prove that $H^{\mathcal{U}} \sqsubset_r \mathcal{WRO}$ when H is Merkle-Damgård based hash constructions such as Prefix-free Merkle-Damgård [12], EMD [5], and MDP [15].

sists of $(\mathcal{RO}, \mathcal{RO}^\dagger, \mathcal{TO}, \mathsf{IC})$, and the interfaces are defined as $\mathcal{WRO}.hon = \mathcal{RO}$ and $\mathcal{WRO}.adv = (\mathcal{RO}, \mathcal{RO}^\dagger, \mathcal{TO}, \mathsf{IC})$.

Appropriateness of $\mathcal{WRO}$. We succeed to bypass the impossible result in [21] by introducing the $\mathcal{WRO}$ model; however, it is non-trivial if previous cryptosystems that are secure for multi-stage games in the $\mathcal{RO}$ model are still secure in the $\mathcal{WRO}$ model. Thus, the next step is to show that there exists a secure cryptosystem for a multi-stage game in the $\mathcal{WRO}$ model. We consider public-key encryption (PKE) (as cryptosystem $\mathcal{C}$) for the Chosen Distribution Attack (CDA) game [1, 2] (as game $\mathcal{S}$). Roughly, we say a PKE scheme is CDA secure if message privacy is preserved even if an adversary can control distributions of messages and randomness in generating the challenge ciphertext. The CDA game captures several flavors of PKE settings (e.g., deterministic PKE (DPKE) [1, 3, 8, 14, 18], hedged PKE (HPKE) [2], and message-locked PKE [4]), and such PKE settings are tools for many practical applications. Thus, our target is to find a CDA secure cryptosystem in the $\mathcal{WRO}$ model.

First, we start with the result in [21]. They showed that any CPA secure PKE scheme in the $\mathcal{RO}$ model can be (redundancy-freely) transformed to an IND-SIM secure PKE scheme in the $\mathcal{RO}$ model via conversion $\mathsf{REwH1}$ [2]. The IND-SIM security is a very weak property that an adversary cannot distinguish between encryptions of chosen messages under chosen randomness and the output of a simulator.[2] We show that any IND-SIM secure [21] PKE scheme in the $\mathcal{RO}$ model is also CDA secure in the $\mathcal{WRO}$ model (Theorem 4). The combination of our theorem and the previous result implies that a CDA secure PKE scheme in the $\mathcal{WRO}$ model can be obtained from any CPA secure PKE scheme in the $\mathcal{RO}$ model.[3]

To prove the CDA security in the $\mathcal{WRO}$ model, we must ensure that the sub oracle $\mathcal{O}^*$ gives no advantage to an adversary in the CDA game. The CDA game consists of two stages, where a first stage adversary $\mathcal{A}_1$ sends no value to a second stage adversary $\mathcal{A}_2$.[4] First, the challenge ciphertext c_β does not leak any information of messages (m_0, m_1) and r even with access to $\mathcal{RO}$. This property is guaranteed by the IND-SIM security. Next, if $\mathcal{RO}^\dagger$ and $\mathcal{RO}^*$ are ideal primitives whose outputs do not leak no information for the inputs, these oracles give no advantage to the adversary. Finally, $\mathcal{A}_1$ might deliver some information about (m_0, m_1) or r via interfaces of IC, $\mathcal{TO}$ and $\mathcal{RO}^\dagger$. $\mathcal{A}_1$ can pose (m_0, m_1) or r (or a related value) to $\mathcal{RO}^\dagger$, E, and D, where E and D are an encryption oracle and a decryption oracle of IC. If $\mathcal{A}_2$ could pose the corresponding output value of $\mathcal{RO}^\dagger$, E, or D to $\mathcal{TO}$, D, or E, $\mathcal{A}_2$ would obtain information from $\mathcal{A}_1$. However, indeed, $\mathcal{A}_2$ cannot find the corresponding output value except negligible probability because of following two reasons: 1) Any meaningful information from $\mathcal{A}_1$ is not obtained from any of c_β, $\mathcal{RO}$, $\mathcal{RO}^\dagger$ and $\mathcal{RO}^*$ as discussed above. 2) Outputs of $\mathcal{RO}^\dagger$, E, and D are uniformly random, and then a possible action of $\mathcal{A}_2$ is randomly guessing these values. Therefore, $\mathcal{TO}$ and IC also give no advantage to the adversary.

[2] This definition is meaningless in the standard model because the encryption algorithm uses no further randomness beyond that input.

[3] From Theorem 2 and 3, the CDA security in the $\mathcal{WRO}$ model is preserved if $\mathcal{WRO}$ is replaced with the ChopMD construction and the FOLSponge construction. Therefore, our result achieves that a CDA secure PKE scheme with such practical hash functions can be obtained from any CPA secure PKE scheme in the $\mathcal{RO}$ model.

[4] In the first stage, an adversary $\mathcal{A}_1$ outputs two messages (m_0, m_1) and a random value r such that the jointed values $m_i \| r$ have sufficient min-entropy. In the second stage, an adversary $\mathcal{A}_2$ receives the challenge ciphertext $c_\beta = \mathcal{E}(m_\beta; r)$ from the game where β is a random value of a single bit, and outputs a bit b, where $\mathcal{E}$ is an encryption function. The adversary wins if $b = \beta$.

1.6 Related Works

There have been some independent studies [13, 16] to consider the indifferentiability framework in multi-stage games. They independently show that for any domain extender H it is impossible to prove $H^{\mathcal{U}} \sqsubset_r \mathcal{RO}$.

Because of the impossibility result, it cannot be guaranteed to securely instantiate $\mathcal{RO}$ by $H^{\mathcal{U}}$ via the reset indifferentiability. Thus, they try to salvage H by relaxing limitations of S and/or $\mathcal{D}$. Conversely, we salvage H by showing instantiability from $\mathcal{WRO}$.

Demay et al. [13] propose a relaxed model that is called *resource-restricted indifferentiability.* This model allows simulator S to have a fixed size state while the reset indifferentiability restrict S to be stateless. That means, adversaries in a multi-stage game can share a fixed size (denoted by parameter s) state. They show that it is possible to securely instantiate $\mathcal{RO}$ by $H^{\mathcal{U}}$ via the resource-restricted indifferentiability. Specifically, they define that F_1 is s-resource-restricted indifferentiable from F_2 (denoted by $F_1 \sqsubset_{rr,s} F_2$) if $\exists S$ with the state size s bit s.t. no $\mathcal{D}$ distinguishes the real world $(F_1.hon, F_1.adv)$ from the ideal world $(F_2.hon, S^{F_2.adv})$. They prove that for any multi-stage game security $\mathcal{S}$ that the size of shared state between adversaries in multi-stage is restricted to equal or lower than s bit, $F_1 \sqsubset_{rr,s} F_2 \Rightarrow \forall \mathcal{C} \in \mathbf{C}\ \mathcal{C}(F_1) \succ_{\mathcal{S}} \mathcal{C}(F_2)$.

They also show a necessary condition of parameter s (i.e., $s = l - m - \log q > 0$) to prove $H^{\mathcal{U}} \sqsubset_{rr,s} \mathcal{RO}$ for any domain extender H, where l is the maximal input length of H, m is the input length of the ideal primitive of H (e.g., compression function) and q is the number of query of S. Their theorem is only valid for the case $s > 0$; that is, their result is still restricted to *specific* multi-stage games. Indeed, unfortunately, their approach *cannot* cover security games that shared state between adversaries in multi-stage is restricted to zero (i.e., $s = 0$). Because the CDA game is the case $s = 0$, they cannot salvage H for the CDA game while our result can do that.

Luykx et al. [16] propose a relaxed model that is called *i-reset indifferentiability.* This model restricts distinguisher $\mathcal{D}$ so that $\mathcal{D}$ is allowed to reset the memory of simulator S only i times while the reset indifferentiability allows $\mathcal{D}$ to reset any times. That means, the number of stages in multi-stage games is equal or lower than i. They define that F_1 is i-reset indifferentiable from F_2 (denoted by $F_1 \sqsubset_{r,i} F_2$) if $\exists S$ which is stateful s.t. no $\mathcal{D}$ distinguishes the real world $(F_1.hon, F_1.adv)$ from the ideal world $(F_2.hon, S^{F_2.adv})$, where $\mathcal{D}$ can reset S up to i times. They prove that for any i'-stage $(1 \le i' \le i)$ game security $\mathcal{S}$, $F_1 \sqsubset_{r,i} F_2 \Rightarrow \forall \mathcal{C} \in \mathbf{C}\ \mathcal{C}(F_1) \succ_{\mathcal{S}} \mathcal{C}(F_2)$.

Unfortunately, they show the impossibility that $H^{\mathcal{U}} \sqsubset_{r,i} \mathcal{RO}$ cannot be proved for *any one-pass hash construction* even if $i = 1$. Hence, their approach *cannot* salvage practical H. On the other hand, our result can salvage important and practical one-pass H such as ChopMD and FOLSponge (Theorems 2 and 3); therefore, our methodology with $\mathcal{WRO}$ is more suitable in a practical sense.

2 Preliminaries

Notations. For two values x, y, $x||y$ is the concatenated value of x and y. For some value y, $x \leftarrow y$ means assigning y to x. When X is a non-empty finite set, we write $x \xleftarrow{\$} X$ to mean that a value is sampled uniformly at random from X and assign to x. $\oplus$ is bitwise exclusive or. $|x|$ is the bit length of x. For sets A and C, $C \xleftarrow{\cup} A$ means assign $A \cup C$ to C. For $l \times r$-bit value M, $div(r, M)$ divides M into r-bit values $(M_1, \ldots, M_l)$ and outputs them where $M_1 \| \cdots \| M_l = M$. For a b-bit value x, $x[i, j]$ is the value from (left) i-th bit to (left) j-th bit where $1 \le i \le j \le b$. For example, let $x = 01101001$, $x[3, 5] = 101$. For a Boolean function F, we denote by "$\exists_1 M$ s.t. $F(M)$ is true" "there exists just a value M such that $F(M)$ is true". Vectors are written in boldface, e.g., $\mathbf{x}$. If $\mathbf{x}$ is a vector then $|\mathbf{x}|$ denotes its length and $\mathbf{x}[i]$ denotes its i-th component for $1 \le i \le |\mathbf{x}|$. $bit_j(\mathbf{x})$ is the left j-th bit of $\mathbf{x}[1] \| \ldots \| \mathbf{x}[|\mathbf{x}|]$.

Throughout this paper, we assume that any algorithm and game is implicitly given a security parameter as input if we do not explicitly state.

Indifferentiability Frameworks [17, 21]. The indifferentiability framework [17] ensures reducibility from one system F_1 to another system F_2 in any single-stage game, where an adversary uses a single state. That is, this framework ensures that the security for any single-stage game is preserved when F_2 is replaced by F_1. This framework ensures the reducibility in any single-stage by proving that information in the F_1 model can be obtained in the F_2 model. This framework deals with two types for information in the F_i model for $i = 1, 2$; Information from an adversarial interface, denoted by $F_i.adv$ to which adversaries have access, and information from an honest interface, denoted by $F_i.hon$ to which honest parties have access. In this framework, the reducibility reflects in a simulation based game, called an indifferentiable game: When considering the reducibility from F_1 to F_2, the advantage of this game is defined as follows.

$$\mathsf{Adv}^{\mathsf{indiff}}_{F_1,F_2,S}(A) = |\Pr[\mathcal{D}^{F_1.hon,F_1.adv} \Rightarrow 1] - \Pr[\mathcal{D}^{F_2.hon,S^{F_2.adv}} \Rightarrow 1]|$$

where S is a simulator which has access to $F_2.adv$ and $\mathcal{D}$ is a distinguisher which has access to left oracle L and right oracle R. The F_1 case is that $(L, R) = (F_1.hon, F_1.adv)$, called Real World. The F_2 case is that $(L, R) = (F_2.hon, S^{F_2.adv})$, called Ideal World. The reducibility from F_1 to F_2 is ensured by showing that there exists a stateful simulator S such that for any $\mathcal{D}$ the indifferentiable advantage is negligible in the security parameter [17].

The reset indifferentiability framework [21] is an extension of the indifferentiability framework and covers any multi-stage game in addition to any single-stage game. A multi-stage game is that the size of the state shared among adversaries are restricted. The restricted situation is covered by dealing with a *stateless* simulator. When considering the reducibility from F_1 to F_2, the advantage of this game is defined as follows.

$$\mathsf{Adv}^{\mathsf{r\text{-}indiff}}_{F_1,F_2,S}(A) = |\Pr[\mathcal{D}^{F_1.hon,F_1.adv} \Rightarrow 1] - \Pr[\mathcal{D}^{F_2.hon,S^{F_2.adv}} \Rightarrow 1]|$$

The reducibility from F_1 to F_2 is ensured by showing that there exists a *stateless* simulator S such that for any $\mathcal{D}$ the indifferentiable advantage is negligible in the security parameter [21]. If there exists such S then F_1 is reset indifferentiable from F_2. More precisely, RSS gave the following theorem.

Theorem 1 (RSS Theorem [21]). *Let G be any game. Let F_1 and F_2 be cryptographic systems. Let S be a stateless simulator. For any adversary $\mathcal{A} = (\mathcal{A}_1, \ldots, \mathcal{A}_m)$, there exist an adversary $\mathcal{B} = (\mathcal{B}_1, \ldots, \mathcal{B}_m)$ and a distinguisher $\mathcal{D}$ such that*

$$\Pr[\mathcal{A} \textit{ wins in } F_1 \textit{ model in } G] \leq \Pr[\mathcal{B} \textit{ wins in } F_2 \textit{ model in } G] + \mathsf{Adv}^{\mathsf{r\text{-}indiff}}_{F_1,F_2,S}(\mathcal{D}).$$

Moreover, $t_{\mathcal{B}_i} \leq t_{\mathcal{A}_i} + q_{\mathcal{A}_i} t_S, q_{\mathcal{B}_i} \leq q_{\mathcal{A}_i} q_S, t_A \leq m + t_G + \sum_{i=1}^{m} q_{G,i} t_{\mathcal{A}_i}, q_A \leq q_{G,0} + \sum_{i=1}^{m} q_{G,i} t_{\mathcal{A}_i}$ where $t_{\mathcal{A}}, t_{\mathcal{B}}, t_{\mathcal{D}}$ are the maximum running times of $\mathcal{A}, \mathcal{B}, \mathcal{D}$; $q_{\mathcal{A}}, q_{\mathcal{B}}$ are the maximum number of queries made by $\mathcal{A}$ and $\mathcal{B}$ in a single execution; and $q_{G,0}, q_{G,1}$ are the maximum number of queries made by G to the private interface and to the adversary.

Definitions of Hash Functions. We give the description of the ChopMD construction [12]. Let h be a compression function which maps a value of $d+n+s$ bits to a value of $n+s$ bits. The ChopMD $\mathrm{ChopMD}^h : \{0,1\}^* \rightarrow \{0,1\}^n$ is defined in Fig. 3. $\mathsf{pad}_c : \{0,1\}^* \rightarrow (\{0,1\}^d)^*$ is an injective padding function such that its inverse is efficiently computable. IV is a constant value of $n + s$ bits.

```
$\mathrm{ChopMD}^h(M)$
$M' \leftarrow \mathsf{pad}_c(M)$;
$(M_1, \ldots, M_i) \leftarrow div(d, M')$;
$x \leftarrow IV$;
for $j = 1, \ldots, i$ do $x \leftarrow h(x \| M_j)$;
return $x[s+1, s+n]$;
```

Fig. 3. Chop Merkle-Damgård

```
Algorithm $\mathrm{FOLSponge}^P(M)$
$M' \leftarrow \mathsf{pad}_S(M)$;
$(M_1, \ldots, M_i) \leftarrow div(n, M')$;
$s = IV$;
for $i = 1, \ldots, i$ do
    $s = P(s \oplus (M_i \| 0^c))$;
return $s[1, n]$;
```

Fig. 4. Sponge

```
$\mathcal{RO}_w^\dagger(M)$
if $\mathsf{F}^\dagger[M] = \perp$ then $\mathsf{F}^\dagger[M] \xleftarrow{\$} \{0,1\}^w$;
return $\mathsf{F}^\dagger[M]$;

$\mathcal{TO}(y)$
if $\exists_1 M$ s.t. $\mathsf{F}^\dagger[M] = y$ then return $M$;
return $\perp$;
```

Fig. 5. $\mathcal{RO}_w^\dagger$ and $\mathcal{TO}$ where $\mathsf{F}^\dagger$ is a (initially everywhere $\perp$) table.

We give the description of the FOLSponge construction [6]. Let P be a permutation of d bits. The FOLSonge $\mathrm{FOLSponge}^P : \{0,1\}^* \to \{0,1\}^n$ is defined in Fig. 4 such that $n < d$.[5] Let $c = d - n$. $\mathsf{pad}_S : \{0,1\}^* \to (\{0,1\}^n)^*$ is an injective padding function such that the last n-bit value is not 0. IV is a constant value of d bits. $IV_1 = IV[1, n]$ and $IV_2 = IV[n+1, d]$. For example, $\mathsf{pad}_S(M) = M \| 1 \| 0^i$ where i is a smallest value such that $|M \| 1 \| 0^i|$ is a multiple of n.

3 Reset Indifferentiability from $\mathcal{WRO}$

RSS [21] proved the impossibility of proving that the ChopMD and the FOLSponge are reset indifferentiable from random oracles. To compensate the impossibility, we change the ideal world from a random oracle to a weakened random oracle ($\mathcal{WRO}$). We define $\mathcal{WRO}$ such that both of the ChopMD and the FOLSponge are reset indifferentiable from $\mathcal{WRO}$s.

3.1 $\mathcal{WRO}$

We define $\mathcal{WRO}$ as $(\mathcal{RO}_n, \mathcal{RO}_v^*, \mathcal{RO}_w^\dagger, \mathcal{TO}, \mathsf{IC}_{a,b})$, where $\mathcal{RO}_n, \mathcal{RO}_v^*, \mathcal{RO}_w^\dagger$ are arbitrary input length random oracles whose output lengths are n bit, v bit, and w bit, respectively, $\mathcal{TO}$ is a trace oracle, and $\mathsf{IC}_{a,b}$ is an ideal cipher with key length a and block length b. The definition of $\mathcal{TO}$ is that for query y to $\mathcal{TO}$, it returns M if $\exists_1 M$ such that a query M to $\mathcal{RO}_w^\dagger$ such that $y = \mathcal{RO}_w^\dagger(M)$ was made, and otherwise it returns $\perp$. Fig. 5 shows the method of implementing a $\mathcal{RO}_w^\dagger$ and a $\mathcal{TO}$. $E : \{0,1\}^a \times \{0,1\}^b \to \{0,1\}^b$ denotes the encryption oracle of $\mathsf{IC}_{a,b}$, and $D : \{0,1\}^a \times \{0,1\}^b \to \{0,1\}^b$ denotes the decryption oracle. The interfaces are defined by $\mathcal{WRO}.hon = \mathcal{RO}_n$ and $\mathcal{WRO}.adv = (\mathcal{RO}_n, \mathcal{RO}_v^*, \mathcal{RO}_w, \mathcal{TO}, \mathsf{IC}_{a,b})$. Note that the parameters (n, v, w, a, b) are defined in each hash function. In Appendix. A, we give a method of implementing $\mathcal{WRO}$.

For a hash function $H^{\mathcal{U}}$ using an ideal primitive $\mathcal{U}$, the advantage of reset indifferentiability from $\mathcal{WRO}$ is defined as follows.

$$\mathsf{Adv}^{\mathsf{r\text{-}indiff}}_{H^{\mathcal{U}}, \mathcal{WRO}, S}(\mathcal{D}) = |\Pr[\mathcal{D}^{H^{\mathcal{U}}, \mathcal{U}} \Rightarrow 1] - \Pr[\mathcal{D}^{\mathcal{WRO}.hon, S^{\mathcal{WRO}.adv}} \Rightarrow 1]|.$$

The RSS theorem ensures that if $H^{\mathcal{U}}$ is reset indifferentiable from a $\mathcal{WRO}$, for any game the security of any cryptosystem is preserved when a $\mathcal{WRO}$ is replaced by $H^{\mathcal{U}}$, where in the $\mathcal{WRO}$

[5] Note that if the output length (denoted l) is smaller than n, the output length is achieved by returning $s[1, l]$.

model adversaries have access to $\mathcal{WRO}.adv$ and the cryptosystem has access to $\mathcal{WRO}.hon$, and for the $H^{\mathcal{U}}$ case, adversaries have access to $\mathcal{U}$ and the cryptosystem has access to $H^{\mathcal{U}}$.

3.2 Reset Indifferentiability for ChopMD

In this proof, we define the parameter of $\mathcal{WRO}$ as $w = s$ and $v = n + s$. Note that $\mathsf{IC}_{a,b}$ is not used. Therefore, $\mathcal{WRO} = (\mathcal{RO}_n, \mathcal{RO}^*_{n+s}, \mathcal{RO}^\dagger_s, \mathcal{TO})$.

Theorem 2. *Let the compression function h be a random oracle. There exists a stateless simulator S such that for any distinguisher $\mathcal{D}$,*

$$\mathsf{Adv}^{\mathsf{r\text{-}indiff}}_{\mathrm{ChopMD}^h,\mathcal{WRO},S}(\mathcal{D}) \leq \frac{q_R(q_R-1)+2\sigma(\sigma+1)}{2^s}$$

where $\mathcal{D}$ can make queries to left oracle $L = \mathrm{ChopMD}^h/\mathcal{RO}_n$ and right oracle $R = h/S$ at most q_L, q_R times, respectively, and l is a maximum number of blocks of a query to L. $\sigma = lq_L + q_R$. S makes at most $4q_R$ queries and runs in time $\mathcal{O}(q_R)$. ♦

An intuition of the proof is shown in Subsection 1.5. The proof for the ChopMD hash function is given in Section 4.

3.3 Reset Indifferentiability for FOLSponge

We define the parameter of $\mathcal{WRO}$ as $w = c$ and $b = d$. We don't care the key size a, since $\mathsf{IC}_{a,b}$ can be regarded as random permutation by fixing a key k^*. We denote $E(k^*, \cdot)$ by a random permutation $\mathcal{P}(\cdot)$ of d bit and $D(k^*, \cdot)$ by its inverse oracle $\mathcal{P}^{-1}(\cdot)$. Note that in this proof, $\mathcal{RO}^*_v$ are not used. Therefore, $\mathcal{WRO} = (\mathcal{RO}_n, \mathcal{RO}^\dagger_c, \mathcal{TO}, \mathcal{P}, \mathcal{P}^{-1})$.

Theorem 3. *Assume that the underlying permutation P is a random permutation and P^{-1} is its inverse oracle. There exists a stateless simulator $S = (S_F, S_I)$ such that for any distinguisher $\mathcal{D}$,*

$$\mathsf{Adv}^{\mathsf{r\text{-}indiff}}_{\mathrm{FOLSponge}^P,\mathcal{WRO},S}(\mathcal{D}) \leq \frac{2\sigma(\sigma+1)+q(q-1)}{2^c} + \frac{\sigma(\sigma-1)+q(q-1)}{2^{d+1}}$$

where $\mathcal{D}$ can make at most q_L, q_F and q_I queries to left $L = \mathrm{FOLSponge}^P/\mathcal{RO}_n$ and right oracles $R_F = P/S_F, R_I = P^{-1}/S_I$. l is a maximum number of blocks of a query to L. $\sigma = lq_L + q_F + q_I$ and $q = q_F + q_I$. S makes at most $4q$ queries and runs in time $\mathcal{O}(q)$. ♦

In the following, we outline why a stateless simulator can be constructed. To simplify the explanation, we omit the padding function of $\mathrm{FOLSponge}^P$. Therefore, queries to L are in $(\{0,1\}^n)^*$. Since $\mathcal{D}$ interacts with (L, R_F, R_I), helpful information for $\mathcal{D}$ is obtained from these oracles. Thus, the S's tasks are to simulate the following two points.

- Simulation of P and P^{-1}: Since in the real world $R_F = P$ and $R_I = P^{-1}$, S must simulate P and P^{-1}.
- Simulation of L-R relation: There is a relation based on the FOLSponge construction among query-response values of L and of R_F in the real world, since $L = \mathrm{FOLSponge}^P$ and $R_F = P$. We consider the following example.
  - $\mathcal{D}$ makes query X_1 ($:= IV \oplus (M_1 \| 0^c)$) to R_F and receives the response Y_1.
  - $\mathcal{D}$ makes query X_2 ($:= Y_1 \oplus (M_2 \| 0^c)$) to R_F and receives the response Y_2.

  In the real world, there are the relations $Y_1[1,n] = L(M_1)$ and $Y_2[1,n] = L(M_1 \| M_2)$.

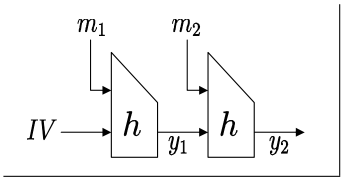

Fig. 6. Figure of Merkle-Damgård

```
$\underline{S(x\|m)}$ where $x_1 = x[1,s], x_2 = x[s+1,n]$ and $|m| = d$
1 $M \leftarrow \mathcal{TO}(x_1)$;
2 if $x = IV$ then
3     $z \leftarrow \mathcal{RO}_n(m)$;
4     $w \leftarrow \mathcal{RO}_s^{\dagger}(m)$;
5 else if $M \neq \perp$ and $x_2 = \mathcal{RO}_n(M)$ then
6     $z \leftarrow \mathcal{RO}_n(M\|m)$;
7     $w \leftarrow \mathcal{RO}_s^{\dagger}(M\|m)$;
8 else $w\|z \leftarrow \mathcal{RO}_{n+s}^{*}(x\|m)$;
9 **return** $w\|z$;
```

Fig. 7. Simulator S

Using $\mathcal{WRO}$, we can construct a stateless simulator which succeeds in these simulations.

- Simulation of P and P^{-1}: S succeeds in this simulation by using $\mathcal{P}$ and $\mathcal{P}^{-1}$; S returns the response of $\mathcal{P}(x)$ for query x, and returns the response of $\mathcal{P}^{-1}(y)$ for query y.
- Simulation of L-R relation: S succeeds in this simulation by using $\mathcal{RO}_c^{\dagger}$ and $\mathcal{TO}$. For example, we consider the above queries by $\mathcal{D}$.
  - For query X_1 to S_F, S_F parses $X_1 = W_1\|IV_2$, $M_1 = W_1 \oplus IV_1$, $Y_1^* := \mathcal{RO}_n(M_1)$, $Y_1' := \mathcal{RO}_c^{\dagger}(M_1)$ and $Y_1 = Y_1^*\|Y_1'$.
  - For query X_2 S_F parses $X_2 = W_2\|Y_1'$, $M_1 = \mathcal{TO}(Y_1')$, $Y_1^* = \mathcal{RO}_n(M_1)$, $M_2 = W_2 \oplus Y_1^*$ and $Y_2 := \mathcal{RO}_n(M_1\|M_2)\|\mathcal{RO}_c^{\dagger}(M_1\|M_2)$.

  These procedures ensure that the relations $Y_1[1,n] = L(M_1)$ and $Y_2[1,n] = L(M_1\|M_2)$ are satisfied.

As a result, we can construct a stateless simulator S which succeeds in the simulations of (P, P^{-1}) and of the L-R relation. Thus we can prove Theorem 3. The proof is given in Appendix B.

4 Proof of Theorem 2

First we define a graph G_{MD}, which is initialized with a single node IV. Edges and nodes in this graph are defined by query-response values to R, which follow the MD structure. The nodes are chaining values and the edges are message blocks. For example, if $(IV, m_1, y_1), (y_1, m_2, y_2)$ are query response values of R, (IV, y_1, y_2) are the nodes of the graph and (m_1, m_2) are the edges. We denote the MD path by $IV \xrightarrow{m_1} y_1 \xrightarrow{m_2} y_2$ or $IV \xrightarrow{m_1\|m_2} y_2$ (Fig. 6 may help to understand the path).

In this proof, the padding function pad_c is removed. Thus queries to L are in $(\{0,1\}^d)^*$. Since the ChopMD with pad_c is the special case of one without pad_c, the security of the ChopMD without pad_c ensures the security of one with pad_c.

We define a stateless simulator S in Fig. 7. Step 8 ensures the simulation of h and Steps 2-7 ensure the simulation of the L-R relation.

4.1 Detail

In the following, for the simulator S in Fig. 7 and any distinguisher $\mathcal{D}$, we evaluate the bound of the reset indiferentiable advantage of ChopMD^h from $\mathcal{WRO}$. To evaluate the bound we consider the following five games. In each game, $\mathcal{D}$ has access to (L, R).

- Game 1 is the ideal world, that is, $(L, R) = (\mathcal{RO}_n, S)$.

- Game 2 is $(L, R) = (\mathcal{RO}_n, S_1)$, where S_1 keeps all query-response pairs. For a query $x\|m$ to S_1, if there is $(x\|m, w\|z)$ in the query response history, then S_1 returns $w\|z$, otherwise, S_1 returns the output of $S(x\|m)$.
- Game 3 is $(L, R) = (L_1, S_1)$, where for a query M to L_1 L_1 makes S_1 queries corresponding with $\mathrm{ChopMD}^{S_1}(M)$ and returns the response of $\mathcal{RO}_n(M)$.
- Game 4 is $(L, R) = (\mathrm{ChopMD}^{S_1}, S_1)$.
- Game 5 is the real world, that is, $(L, R) = (\mathrm{ChopMD}^h, h)$.

Let G_i be an event that $\mathcal{D}$ outputs 1 in Game i. We thus have that

$$\mathsf{Adv}^{\text{r-indiff}}_{\mathrm{ChopMD}^h,\mathcal{WRO},S}(\mathcal{D}) \leq \sum_{i=1}^{4} |\Pr[G_i] - \Pr[G_{i+1}]| \leq \frac{q_R(q_R-1) + 2\sigma(\sigma+1)}{2^s}.$$

In the following, we justify the above bound by evaluating each difference.

Game 1 ⇒ Game 2. From Game 1 to Game 2, we change R from S to S_1 where S_1 records query response values, while S does not record them. The query-response history ensures that in Game 2 if a query $x\|m$ to S_1 was made and y was responded, for the repeated query $x\|m$ to S_1 the same value y is responded, while in Game 1 there is a case that for some repeated query $x\|m$ to S_1 where y was responded, a distinct value y^* ($\neq y$) is responded. The difference $|\Pr[G_1] - \Pr[G_2]|$ is thus bounded by the probability that in Game 1 a different value is responded. We call the event "**Diff**". Since the procedure to define an output of S is controlled by $\mathcal{TO}$ (See the steps 2, 5, and 8), the event **Diff** relies on an output of $\mathcal{TO}$. Thus, if **Diff** occurs, for some repeated query to $\mathcal{TO}$ the response is changed. More precisely, if **Diff** occurs, the following event occurs.

- For a query y to $\mathcal{TO}$, w was responded, and then for the repeated query a different value w^* is responded. From the definition of $\mathcal{TO}$, there are two cases for (w, w^*).
  - $\mathbf{Diff}_1$: $w =\perp$ and $w^* \neq\perp$.
  - $\mathbf{Diff}_2$: $w \neq\perp$ and $w^* =\perp$.

We thus have that

$$|\Pr[G_1] - \Pr[G_2]| \leq \Pr[\mathbf{Diff}_1] + \Pr[\mathbf{Diff}_2] \leq \frac{q_R(q_R-1)}{2^s}.$$

We justify the bound as follows.

First we bound the probability of $\Pr[\mathbf{Diff}_1]$. Since the response w of the first query is $\perp$, when the first query is made, the query w^* to $\mathcal{RO}^\dagger_s$ such that $y = \mathcal{RO}^\dagger_s(w^*)$ was not made. Since the response w^* of the repeated query is not $\perp$, when the repeated query is made, the query w^* to $\mathcal{RO}^\dagger_s$ was made such that $y = \mathcal{RO}^\dagger_s(w^*)$. Therefore, first y is defined. Second, the output of $\mathcal{RO}^\dagger_s(w^*)$ is defined. Thus, $\Pr[\mathbf{Diff}_1]$ is bounded by the probability that the response of $\mathcal{RO}^\dagger_s(w^*)$, which is an s-bit random value, hits the value y. Since the numbers of queries to $\mathcal{RO}^\dagger_s$ and $\mathcal{TO}$ are at most q_R times,

$$\Pr[\mathbf{Diff}_1] \leq \sum_{i=1}^{q_R} \frac{i-1}{2^s} \leq \frac{q_R(q_R-1)}{2^{s+1}}.$$

Next we bound the probability of $\Pr[\mathbf{Diff}_2]$. Since the response w of the first query is not $\perp$, when the first query is made, the query w to $\mathcal{RO}^\dagger_s$ was made such that $y = \mathcal{RO}^\dagger_s(w)$. Since the response w^* of the repeated query is $\perp$, when the repeated query is made, a query w' to $\mathcal{RO}^\dagger_s$ was made such that $w \neq w'$ and $\mathcal{RO}_s^\dagger(w) = \mathcal{RO}_s^\dagger(w')$. Therefore, $\Pr[\mathbf{Diff}_2]$ is bounded by the collision probability of $\mathcal{RO}_s^\dagger$. We thus have that

$$\Pr[\mathbf{Diff}_2] \leq \sum_{i=1}^{q_R} \frac{i-1}{2^s} \leq \frac{q_R(q_R-1)}{2^{s+1}}.$$

Game 2 ⇒ Game 3. From Game 2 to Game 3, we change L from $\mathcal{RO}_n$ to L_1 where in Game 3 L makes additional queries to R corresponding with the calculation of $\mathrm{ChopMD}^{S_1}(M)$. Note that $\mathcal{D}$ cannot directly observe the additional query response values but can observe those by making the queries to R. So we have to show that in Game 3 the additional queries by L don't affect $\mathcal{D}$'s behavior. We ensure this by Lemma 1 where in Game j, for any MD path $IV \xrightarrow{M} z$, $z = \mathcal{RO}_s^\dagger(M)\|\mathcal{RO}_n(M)$ unless Bad_j occurs. By Lemma 1, in both games, unless the bad event occurs, all responses to R are defined by the same queries to $\mathcal{RO}_s^\dagger$ and to $\mathcal{RO}_n$. Namely, in Game 3, the responses of the additional queries to R which $\mathcal{D}$ observes are chosen from the same distribution as in Game 2 unless the bad event occurs. Thus, the difference $|\Pr[G_2] - \Pr[G_3]|$ is bounded by the probability of occurring the bad event.

First we define the bad event. Let T_i be a list which records $(x_t[1,s], y_t[1,s])$ for $t = 1, \ldots, i-1$ where $(x_t\|m_t, y_t)$ is a t-th query response pair of S where $y_t = S(x_t\|m_t)$.

- Bad_j is that in Game j for some i-th query $x_i\|m_i$ to S, the response y_i is such that $y_i[1,s]$ collides with some value in $T_i \cup \{x_i[1,s]\} \cup \{IV[1,s]\}$.

Note that since all outputs of S_1 are defined by using S, we deal with S instead of S_1.

Next we give Lemma 1 as follows. Note that Lemma 1 is also used when evaluating the difference between Game 3 and Game 4.

Lemma 1. *In Game j, unless Bad_j occurs, for any MD path $IV \xrightarrow{M} y$ $y = \mathcal{RO}_s^\dagger(M)\|\mathcal{RO}_n(M)$.* ♦

Proof of Lemma 1. Assume that Bad_j does not occur. We show that for any MD path $IV \xrightarrow{M} y$, $y = \mathcal{RO}_s^\dagger(M)\|\mathcal{RO}_n(M)$. Let $(x_1\|m_1, y_1), \ldots, (x_t\|m_t, y_t)$ be query response pairs to S which correspond with the MD path where $x_1 = IV$, $x_i = y_{i-1}$ $(i = 2, \ldots, t)$, $y_t = y$, and $M = m_1\|\ldots\|m_t$.

When $t = 1$, $y = \mathcal{RO}_s^\dagger(M)\|\mathcal{RO}_n(M)$ (see Steps 2-4).

We consider the case that $t \geq 2$.

Since Bad_j does not occur, the following case does not occur; for some $i \in \{1, \ldots, t-1\}$, $(x_i\|m_i, y_i)$ is defined after $(x_{i+1}\|m_{i+1}, y_{i+1})$ was defined. So $(x_1\|m_1, y_1), \ldots, (x_t\|m_t, y_t)$ are defined by this order.

Since Bad_j does not occur, no collision of outputs of $\mathcal{RO}_s^\dagger$ occurs. Therefore, when the query $S_1(x_t\|m_t)$ is made, the pair $(m_1\|\ldots\|m_{j-1}, y_{t-1})$ has been recorded in the table $\mathsf{F}^\dagger$ of $\mathcal{RO}_s^\dagger$, that is, $\mathsf{F}^\dagger[m_1\|\ldots\|m_{t-1}] = y_{t-1} = x_t$.

Since Bad_j does not occur, no collision of outputs of $\mathcal{RO}_s^\dagger$ occurs. Therefore, there is no value M^* such that $M^* \neq m_1\|\ldots\|m_{t-1}$ and $\mathsf{F}^\dagger[M^*] = x_t$.

Thus, for the query $x_t\|m_t$ to S, S makes the query $x_t[1,s]$ to $\mathcal{TO}$, receives $m_1\|\ldots\|m_{t-1}$ (Step 1), and returns the response y_t such that $y_t = \mathcal{RO}_s^\dagger(M)\|\mathcal{RO}_n(M)$ (Step 6).

□

By Lemma 1, we can bound the difference $|\Pr[G_2] - \Pr[G_3]|$ as follows.

$$\begin{aligned}|\Pr[G_2] - \Pr[G_3]| \leq & |\Pr[G_2|Bad_2]\Pr[Bad_2] + \Pr[G_2|\neg Bad_2]\Pr[\neg Bad_2] \\ & \quad - (\Pr[G_3|Bad_3]\Pr[Bad_3] + \Pr[G_3|\neg Bad_3]\Pr[\neg Bad_3])| \\ \leq & |\Pr[G_2|\neg Bad_2](\Pr[Bad_3] - \Pr[Bad_2]) \\ & \quad + (\Pr[G_2|Bad_2]\Pr[Bad_2] - \Pr[G_3|Bad_3]\Pr[Bad_3])| \\ \leq & \max\{\Pr[Bad_2], \Pr[Bad_3]\} \leq \frac{\sigma(\sigma+1)}{2^s}\end{aligned}$$

where $\Pr[G2|\neg Bad_2] = \Pr[G3|\neg Bad_3]$ from Lemma 1. Finally we justify the bound. The left s-bit values of all outputs of S_1 are uniformly chosen at random from $\{0,1\}^s$. The probability of occurring the bad event is that for some i-th query to S the left s-bit value of the response, which is a random value, hits some of $T_i \cup \{x_i[1,s]\} \cup \{IV[1,s]\}$. We thus have that

$$\Pr[Bad_2] \leq \sum_{i=1}^{q_R} \frac{2(i-1)+2}{2^s} = \frac{q_R(q_R+1)}{2^s},\ \Pr[Bad_3] \leq \sum_{i=1}^{\sigma} \frac{2(i-1)+2}{2^s} = \frac{\sigma(\sigma+1)}{2^s}$$

where S_1 is called at most q_R times in Game 2 and σ times in Game 3.

Game 3 ⇒ Game 4. From Game 3 to Game 4, we change L where in Game 3 $L(M) = \mathcal{RO}_n(M)$, while in Game 4 $L(M) = \text{ChopMD}^{S_1}(M)$. Therefore, the modification does not change $\mathcal{D}$'s behavior iff in Game 4 $\text{ChopMD}^{S_1}(M) = \mathcal{RO}_n(M)$. Since Lemma 1 ensures that for any MD path $IV \xrightarrow{M} z$, $z = \mathcal{RO}_s^{\dagger}(M) \| \mathcal{RO}_n(M)$ unless the bad event Bad_4 occurs, the modification does not change $\mathcal{D}$'s behavior. Thus the difference $|\Pr[G_3] - \Pr[G_4]|$ is bounded by the probability of occurring Bad_4. Since S_1 is called at most σ times, we have that

$$|\Pr[G_3] - \Pr[G_4]| \leq \Pr[Bad_4] \leq \frac{\sigma(\sigma+1)}{2^s}.$$

Game 4 ⇒ Game 5. From Game 4 to Game 5, we change R from S_1 to h. Since outputs of S_1 are uniformly chosen at random from $\{0,1\}^{n+s}$, the modification of R does not affect $\mathcal{D}$'s behavior. We thus have that $\Pr[G_4] = \Pr[G_5]$. □

5 Multi-Stage Security in the $\mathcal{WRO}$ Model

In this section, we show appropriateness of our $\mathcal{WRO}$ methodology. We construct a (non-adaptive) CDA secure [2] PKE scheme in the $\mathcal{WRO}$ model. Specifically, we show that if a PKE scheme satisfies an weak security (i.e., IND-SIM security [21]) in the $\mathcal{RO}$ model, then it is also CDA secure in the $\mathcal{WRO}$ model.

An IND-SIM secure PKE in the $\mathcal{RO}$ model is easily obtained by applying a known technique [21] that any CPA secure PKE scheme can be converted into IND-SIM secure by using EwH [1] and REwH1 [2] in the $\mathcal{RO}$ model. Therefore, our result implies that a very large class of PKE schemes is CDA secure in the $\mathcal{WRO}$ model (e.g., factoring-based, Diffie-Hellman-based, lattice-based, etc.).

Furthermore, our result in Section 3 guarantees to instantiate $\mathcal{WRO}$ by ChopMD or FOLSponge. Hence, finally, we have that any CPA secure PKE in the $\mathcal{RO}$ model can be converted into CDA secure with ChopMD or FOLSponge. While the previous work [21] showed CDA secure PKE schemes only with the specific NMAC hash function, our work achieves CDA secure PKE schemes with large class of hash functions (i.e., ChopMD and FOLSponge).

```
$\mathrm{CDA}^{\mathcal{A}_1,\mathcal{A}_2}_{\mathcal{AE},F}$
$\beta \xleftarrow{\$} \{0,1\}$
$(pk, sk) \xleftarrow{\$} \mathsf{Gen}$
$(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r}) \leftarrow \mathcal{A}_1^{F.adv}$
$\mathbf{c} \leftarrow \mathsf{Enc}^{F.hon}(pk, \mathbf{m}_\beta; \mathbf{r})$
$\beta' \leftarrow \mathcal{A}_2^{F.adv}(pk, \mathbf{c})$
return $(\beta = \beta')$
```

```
$\text{IND-SIM}^{\mathcal{B}}_{\mathcal{AE},\mathcal{S},F}$
$\beta \xleftarrow{\$} \{0,1\}$
$(pk, sk) \xleftarrow{\$} \mathsf{Gen}$
$\beta' \leftarrow \mathcal{B}^{\mathsf{RoS},F.adv}(pk)$
return $(\beta = \beta')$

$\mathsf{RoS}(m, r)$
If $\beta = 1$ then return $\mathsf{Enc}^{F.hon}(pk, m; r)$
Otherwise return $\mathcal{S}^{F.hon}(pk, |m|)$
```

Fig. 8. CDA game and IND-SIM game

5.1 CDA Secure PKE in the $\mathcal{WRO}$ Model

Public Key Encryption (PKE). A public key encryption scheme $\mathcal{AE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ consists of three algorithms. Key generation algorithm Gen outputs public key pk and secret key sk. Encryption algorithm Enc takes public key pk, plaintext m, and randomness r, and outputs ciphertext c. Decryption algorithm Dec takes secret key sk and ciphertext c, and outputs plaintext m or distinguished symbol $\perp$. For vectors $\mathbf{m}, \mathbf{r}$ with $|\mathbf{m}| = |\mathbf{r}| = l$ which is the size of vectors, we denote by $\mathsf{Enc}(pk, \mathbf{m}; \mathbf{r})$ the vector $(\mathsf{Enc}(pk, \mathbf{m}[1]; \mathbf{r}[1]), \ldots, \mathsf{Enc}(pk, \mathbf{m}[l]; \mathbf{r}[l]))$. We say that $\mathcal{AE}$ is deterministic if Enc is deterministic.

CDA Security. We explain the CDA security (we quote the explanation of the CDA security in [21]). Fig. 8 illustrates the non-adaptive CDA game for a PKE scheme $\mathcal{AE}$ using a functionality F. This notion captures the security of a PKE scheme when randomness $\mathbf{r}$ used in encryption may not be a string of uniform bits. For the remainder of this section, fix a randomness length $\rho \geq 0$ and a plaintext length $\omega > 0$. An (μ, ν)-mmr-source $\mathcal{M}$ is a randomized algorithm that outputs a triple of vector $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$ such that $|\mathbf{m}_0| = |\mathbf{m}_1| = |\mathbf{r}| = \nu$, all components of $\mathbf{m}_0$ and $\mathbf{m}_1$ are bit strings of length ω, all components of $\mathbf{r}$ are bit strings of length ρ, and $(\mathbf{m}_\beta[i], \mathbf{r}[i]) \neq (\mathbf{m}_\beta[j], \mathbf{r}[j])$ for all $1 \leq i < j \leq \nu$ and all $\beta \in \{0,1\}$. Moreover, the source has min-entropy μ, meaning $\Pr[(\mathbf{m}_\beta[i], \mathbf{r}[i]) = (m', r') | (\mathbf{m}_0, \mathbf{m}_1, \mathbf{r}) \leftarrow \mathcal{M}] \leq 2^{-\mu}$ for all $\beta \in \{0,1\}$, all $1 \leq i \leq \nu$, and all (m', r'). A CDA adversary $\mathcal{A}_1, \mathcal{A}_2$ is a pair of procedures, the first of which is a (μ, ν)-mmr-source. The CDA advantage for a CDA adversary $\mathcal{A}_1, \mathcal{A}_2$ against scheme $\mathcal{AE}$ using a functionality F is defined by

$$\mathsf{Adv}^{\mathrm{cda}}_{\mathcal{AE},F}(\mathcal{A}_1, \mathcal{A}_2) = 2 \cdot \Pr[\mathrm{CDA}^{\mathcal{A}_1,\mathcal{A}_2}_{\mathcal{AE},F} \Rightarrow \mathsf{true}] - 1.$$

As noted in [2], in the RO model, mmr-sources have access to the RO. In this setting, the min-entropy requirement is independent of the coins used by the RO, meaning the bound must hold for any fixed choice of function as the RO. If this condition is removed, one can easily break the CDA security (i.e., $\mathcal{A}_1$ and $\mathcal{A}_2$ can easily share the messages $(\mathbf{m}_1, \mathbf{m}_2, \mathbf{r})$) for any cryptosystem using any indifferentiable hash function.

IND-SIM Security. The IND-SIM security is a special notion for PKE schemes. It captures that an adversary cannot distinguish outputs from the encryption algorithm and from a simulator $\mathcal{S}$ even if the adversary can choose plaintext and randomness. Fig. 8 shows the IND-SIM game. We define the IND-SIM advantage of an adversary $\mathcal{B}$ by

$$\mathsf{Adv}^{\text{ind-sim}}_{\mathcal{AE},\mathcal{S},F}(\mathcal{B}) = 2 \cdot \Pr[\text{IND-SIM}^{\mathcal{B}}_{\mathcal{AE},F} \Rightarrow \mathsf{true}] - 1.$$

As noted in [21], in the standard model this security goal is not achievable because $\mathcal{AE}$ uses no randomness beyond that input. In the RO model, we will use it when the adversary does not make any RO queries. A variety of PKE schemes is shown to satisfy IND-SIM security in the RO model.

CDA Security in the $\mathcal{WRO}$ Model. The following theorem shows that for any PKE scheme the non-adaptive CDA security in the $\mathcal{WRO}$ model is obtained from IND-SIM security in the RO model.

Theorem 4. *Let $\mathcal{AE}$ be a PKE scheme. Let $(\mathcal{A}_1, \mathcal{A}_2)$ be a CDA adversary in the $\mathcal{WRO}$ model making at most $q_{\mathcal{RO}}, q_{\mathcal{RO}^*}, q_{\mathcal{RO}^\dagger}, q_{\mathcal{TO}}, q_E, q_D$ queries to $\mathcal{RO}_n, \mathcal{RO}^*_v, \mathcal{RO}^\dagger_w, \mathcal{TO}, \mathsf{IC}_{a,b} = (E, D)$. For any simulator $\mathcal{S}$ there exists an IND-SIM adversary $\mathcal{B}$ such that*

$$\begin{aligned}\mathsf{Adv}^{\text{cda}}_{\mathcal{AE},\mathcal{WRO}}(\mathcal{A}_1,\mathcal{A}_2) \le \mathsf{Adv}^{\text{ind-sim}}_{\mathcal{AE},\mathcal{S},\mathcal{RO}_n}(\mathcal{B}) + q_{\mathcal{RO}} \cdot \mathsf{maxpk}_{\mathcal{AE}} + \frac{q_{\mathcal{RO}} + 4q^2_{\mathcal{RO}^*} + 4q^2_{\mathcal{RO}^\dagger}}{2^\mu} \\ + max\left\{\frac{4q^2_{\mathcal{TO}}}{2^\mu}, \frac{4q^2_{\mathcal{TO}}}{2^w}\right\} + max\left\{\frac{4q^2_E + 4q^2_D}{2^\mu}, \frac{4q^2_E + 4q^2_D}{2^b}\right\}.\end{aligned}$$

$\mathcal{B}$ makes no RO queries, makes ν RoS-queries, and runs in time that of $(\mathcal{A}_1, \mathcal{A}_2)$ plus $\mathcal{O}(q_{\mathcal{RO}} + q_{\mathcal{RO}^*} + q_{\mathcal{RO}^\dagger} + q_{\mathcal{TO}} + q_E + q_D)$. $\mathsf{maxpk}_{\mathcal{AE}}$ is the maximum public key collision probability defined as $\mathsf{maxpk}_{\mathcal{AE}} = \max_{\gamma \in \{0,1\}^*} \Pr[pk = \gamma : (pk, sk) \xleftarrow{\$} \mathsf{Gen}]$. ♦

The proof outline is as follows: First, we start with game $\mathbf{G}_0$ which is exactly the same game as the CDA game in the $\mathcal{WRO}$ model. Secondly, we transform $\mathbf{G}_0$ to game $\mathbf{G}_1$ so that $\mathcal{RO}_n$ returns a random value when $\mathcal{A}_1$ poses a message that is posed to $\mathcal{RO}_n$ by Enc to generate the challenge ciphertext. In game $\mathbf{G}_1$, outputs of $\mathcal{RO}_n$ does not contain any information about computations to generate the challenge ciphertext for $\mathcal{A}_1$. Thirdly, we transform $\mathbf{G}_1$ to game $\mathbf{G}_2$ so that ciphertext $\mathbf{c}$ is generated from a simulator $\mathcal{S}$ in the IND-SIM game. In game $\mathbf{G}_2$, ciphertext $\mathbf{c}$ does not contain any information about outputs of $\mathcal{A}_1$. Thus, $\mathcal{A}_1$ cannot hand over any information to $\mathcal{A}_2$ with $\mathbf{c}$. Fourthly, we transform $\mathbf{G}_2$ to game $\mathbf{G}_3$ so that the table of inputs and outputs of each oracle in $\mathcal{WRO}$ (except $\mathcal{RO}_n$) for $\mathcal{A}_1$ is independent of the table for $\mathcal{A}_2$ according to the output of $\mathcal{A}_1$. In game $\mathbf{G}_3$, queries to sub-oracles for $\mathcal{A}_2$ does not contain any information about the output of $\mathcal{A}_1$, and $\mathcal{A}_1$ cannot hand over any information to $\mathcal{A}_2$ with sub-oracles. Finally, we transform $\mathbf{G}_3$ to game $\mathbf{G}_4$ so that $\mathcal{RO}_n$ returns a random value when $\mathcal{A}_2$ poses a message that is posed to $\mathcal{RO}_n$ by Enc to generate the challenge ciphertext. In game $\mathbf{G}_4$, outputs of $\mathcal{RO}_n$ does not contain any information about computations to generate the challenge ciphertext for $\mathcal{A}_2$. Thus, the advantage of $\mathcal{A}_2$ in $\mathbf{G}_4$ is nothing.

The proof of Theorem 4 is shown in Appendix C.

5.2 Another Secure Cryptosystem in the $\mathcal{WRO}$ Model

In addition to the PKE setting, we consider the ID-based encryption (IBE) setting. Specifically, we show a generic construction of IBE, called $\mathsf{IDREwH1}$ which is an analogy of $\mathsf{REwH1}$, and is non-adaptive ID-based CDA (ID-CDA) secure in the $\mathcal{WRO}$ model if underlying IBE scheme is ID-CPA secure in the RO model. Therefore, any ID-CPA secure IBE in the RO model can be generically converted into ID-CDA secure IBE in the $\mathcal{WRO}$ model.

The detail of the result on IBE is shown in Appendix D.

Acknowledgments

We thank Arno Mittelbach for pointing out typos of Fig. 7 and Fig. 14.

References

1. Mihir Bellare, Alexandra Boldyreva, and Adam O'Neill. Deterministic and Efficiently Searchable Encryption. In *CRYPTO*, volume 4622 of *Lecture Notes in Computer Science*, pages 535–552. Springer, 2007.
2. Mihir Bellare, Zvika Brakerski, Moni Naor, Thomas Ristenpart, Gil Segev, Hovav Shacham, and Scott Yilek. Hedged public-key encryption: How to protect against bad randomness. In *ASIACRYPT*, volume 5912 of *LNCS*, pages 232–249. Springer, 2009.
3. Mihir Bellare, Marc Fischlin, Adam O'Neill, and Thomas Ristenpart. Deterministic Encryption: Definitional Equivalences and Constructions without Random Oracles. In *CRYPTO*, volume 5157 of *LNCS*, pages 360–378. Springer, 2008.
4. Mihir Bellare, Sriram Keelveedhi, and Thomas Ristenpart. Message-Locked Encryption and Secure Deduplication. In *EUROCRYPT 2013*, 2013.
5. Mihir Bellare and Thomas Ristenpart. Multi-Property-Preserving Hash Domain Extension and the EMD Transform. In *ASIACRYPT*, volume 4284 of *Lecture Notes in Computer Science*, pages 299–314. Springer, 2006.
6. Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. On the Indifferentiability of the Sponge Construction. In *EUROCRYPT*, pages 181–197, 2008.
7. Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. The Keccak SHA-3 submission. Submission to NIST (Round 3). 2011.
8. Alexandra Boldyreva, Serge Fehr, and Adam O'Neill. On Notions of Security for Deterministic Encryption, and Efficient Constructions without Random Oracles. In *CRYPTO*, volume 5157 of *LNCS*, pages 335–359. Springer, 2008.
9. Dan Boneh and Matthew K. Franklin. Identity-Based Encryption from the Weil Pairing. In *CRYPTO 2001*, pages 213–229, 2001.
10. Ran Canetti, Shai Halevi, and Jonathan Katz. A Forward-Secure Public-Key Encryption Scheme. In *EUROCRYPT 2003*, pages 255–271, 2003.
11. Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-Ciphertext Security from Identity-Based Encryption. In *EUROCRYPT 2004*, pages 207–222, 2004.
12. Jean-Sébastien Coron, Yevgeniy Dodis, Cécile Malinaud, and Prashant Puniya. Merkle-Damgård Revisited: How to Construct a Hash Function. In *CRYPTO*, volume 3621 of *Lecture Notes in Computer Science*, pages 430–448. Springer, 2005.
13. Gregory Demay, Peter Gazi, Martin Hirt, and Ueli Maurer. Resource-restricted indifferentiability. In *EUROCRYPT*, volume 7881 of *Lecture Notes in Computer Science*, pages 665–684. Springer, 2013.
14. Benjamin Fuller, Adam O'Neill, and Leonid Reyzin. A Unified Approach to Deterministic Encryption: New Constructions and a Connection to Computational Entropy. In *TCC2012*, pages 582–599, 2012.
15. Shoichi Hirose, Je Hong Park, and Aaram Yun. A Simple Variant of the Merkle-Damgård Scheme with a Permutation. In *ASIACRYPT*, volume 4833 of *Lecture Notes in Computer Science*, pages 113–129. Springer, 2007.
16. Atul Luykx, Elena Andreeva, Bart Mennink, and Bart Preneel. Impossibility results for indifferentiability with resets. In *ePrint 2012/644*, 2012.
17. Ueli M. Maurer, Renato Renner, and Clemens Holenstein. Indifferentiability, Impossibility Results on Reductions, and Applications to the Random Oracle Methodology. In *TCC*, volume 2951 of *Lecture Notes in Computer Science*, pages 21–39. Springer, 2004.
18. Ilya Mironov, Omkant Pandey, Omer Reingold, and Gil Segev. Incremental Deterministic Public-Key Encryption, (Full Version in ePrint 2012/047). In *EUROCRYPT*, 2012.
19. National Institute of Standards and Technology. Cryptographic Hash Algorithm Competition. `http://csrc.nist.gov/groups/ST/hash/sha-3/winner_sha-3.html`.
20. National Institute of Standards and Technoloty. FIPS PUB 180-4 Secure Hash Standard. In *FIPS PUB*, 2012.
21. Thomas Ristenpart, Hovav Shacham, and Thomas Shrimpton. Careful with Composition: Limitations of the Indifferentiability Framework. In *EUROCRYPT (Full Version: ePrint 2011/339)*, volume 6632 of *Lecture Notes in Computer Science*, pages 487–506. Springer, 2011.
22. Victor Shoup. Sequences of games: a tool for taming complexity in security proofs. In *Cryptology ePrint Archive: 2004/332*, 2004. `http://eprint.iacr.org/2004/332`.

```
$\underline{\mathcal{RO}_n(M)}$
if $\mathsf{F}[M] = \perp$, $\mathsf{F}[M] \overset{\$}{\leftarrow} \{0,1\}^n$;
return $\mathsf{F}[M]$;

$\underline{\mathcal{RO}^*_v(M)}$
If $\mathsf{F}^*[M] = \perp$, $\mathsf{F}^*[M] \overset{\$}{\leftarrow} \{0,1\}^v$;
return $\mathsf{F}^*[M]$;

$\underline{\mathcal{RO}^\dagger_w(M)}$
if $\mathsf{F}^\dagger[M] = \perp$ then $\mathsf{F}^\dagger[M] \overset{\$}{\leftarrow} \{0,1\}^w$;
return $\mathsf{F}^\dagger[M]$;

$\underline{\mathcal{TO}(y)}$
if $\exists_1 M$ s.t. $\mathsf{F}^\dagger[M] = y$ then return $M$;
return $\perp$;

$\underline{E(k,x)}$
if $\mathsf{E}[k,x] = \perp$, $y \overset{\$}{\leftarrow} \{0,1\}^b \backslash T^+[k]$;
$Update(k,x,y)$;
return $\mathsf{E}[k,x]$;

$\underline{D(y)}$
if $\mathsf{D}[k,y] = \perp$, $x \overset{\$}{\leftarrow} \{0,1\}^b \backslash T^-[k]$;
$Update(k,x,y)$;
return $\mathsf{D}[k,y]$;
```

Fig. 9. Weakened Random Oracle $\mathcal{WRO}$

A Implementation of $\mathcal{WRO}$

Fig. 9 shows the method of implementing a $\mathcal{WRO}$. $\mathcal{RO}_n$ is shown in Fig. 5 (Left) where F is a (initially everywhere $\perp$) table. $\mathcal{RO}^*_v$ is shown in Fig. 5 (Left) where F^* is a (initially everywhere $\perp$) table. $\mathcal{RO}^\dagger_w$ and $\mathcal{TO}$ are shown in Fig. 5 (Center) where $\mathsf{F}^\dagger$ is a (initially everywhere $\perp$) table. $\mathsf{IC}_{a,b}$ can be implemented as Fig. 5 (Right) where E and D are (initially everywhere $\perp$) tables where for the query $E(k,x)$ (resp. $D(k,y)$) the output is recored in $\mathsf{E}[k,x]$ (resp. $\mathsf{D}[k,y]$). $T^+[k]$ and $T^-[k]$ are (initially empty) tables which store all values of $\mathsf{E}[k,\cdot]$ and $\mathsf{D}[k,\cdot]$, respectively. $Update(k,x,y)$ is the procedure wherein the tables $\mathsf{E}, \mathsf{D}, T^+[k]$ and $T^-[k]$ are updated as $\mathsf{E}[k,x] \leftarrow y, \mathsf{D}[k,y] \leftarrow x, T^+[k] \overset{\cup}{\leftarrow} \{y\}$ and $T^-[k] \overset{\cup}{\leftarrow} \{x\}$.

B Proof of Theorem 3

We define a graph G_S, which is initialized with the single node IV. Edges and nodes in this graph are defined by query response values to R_F and R_I which follow the Sponge structure. The nodes are chaining values and the edges are message blocks. For example, if $(X_1,Y_1),(X_2,Y_2)$ are query response values of R_F or R_I such that $X_1[n+1,d] = IV_2$ and $Y_1[n+1,d] = X_2[n+1,d]$ then IV, Y_1, Y_2 are the nodes of G_S and M_1, M_2 are the edges where $M_1 = IV_1 \oplus X_1[1,n]$ and $M_2 = Y_1[1,n] \oplus X_2[1,n]$. We denote the path by $IV \xrightarrow{M_1} Y_1 \xrightarrow{M_2} Y_2$ or $IV \xrightarrow{M_1||M_2} Y_2$ (Fig. 10 may help to understand the graph). We call a path following the Sponge structure "Sponge path".

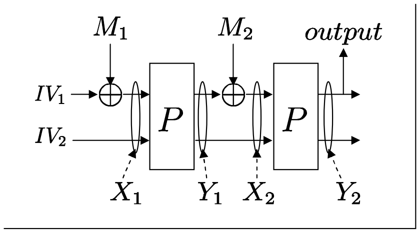

Fig. 10. Figure of Sponge

In this proof, we omit the padding function pad_S. Thus queries to L are in $(\{0,1\}^n)^*$. Note that the FOLSponge with pad_S is the special case of one without pad_S. Thus the security of the FOLSponge without pad_S ensures the security of one with pad_S.

```
$\underline{S_F(X)}$ where $x = X[1,n]$, $y = Y[n+1,d]$
1 $M \leftarrow \mathcal{TO}(y)$;
2 if $y = IV_2$ then
3     $z \leftarrow \mathcal{RO}_n(x \oplus IV_1)$; $w \leftarrow \mathcal{RO}^{\dagger}_c(x \oplus IV_1)$;
4 else if $M \neq \perp$ then
5     $m \leftarrow x \oplus \mathcal{RO}_n(M)$;
6     $z \leftarrow \mathcal{RO}_n(M||m)$; $w \leftarrow \mathcal{RO}^{\dagger}_c(M||m)$;
7 else $z||w \leftarrow \mathcal{P}(x||y)$;
8 return $z||w$;
```

```
$\underline{S_I(Y)}$ where $z = Y[1,n]$, $w = Y[n+1,d]$
1 $M \leftarrow \mathcal{TO}(w)$;
2 if $M \neq \perp$ and $|M| = n$ then
3     $x \leftarrow IV_1 \oplus M$; $y \leftarrow IV_2$;
4 if $M \neq \perp$ and $|M| > n$ then
5     let $M = M^*||m$ ($|m| = n$);
6     $x \leftarrow m \oplus \mathcal{RO}_n(M)$; $y \leftarrow \mathcal{RO}^{\dagger}_c(M^*)$;
7 else $x||y \leftarrow \mathcal{P}^{-1}(z||w)$;
8 return $x||y$;
```

Fig. 11. Simulator S_F (left) and S_I (right)

```
$\underline{\mathcal{P}_1(X)}$
1 if $\exists(j,X,Y) \in \mathcal{Q}$ then return $Y$;
2 $Y \xleftarrow{\$} \{0,1\}^d$; $\mathcal{Q} \xleftarrow{\cup} (t,X,Y)$; $t \leftarrow t+1$;
3 return $Y$;
```

```
$\underline{\mathcal{P}_1^{-1}(X)}$
1 if $\exists(j,X,Y) \in \mathcal{Q}$ then return $X$;
2 $X \xleftarrow{\$} \{0,1\}^d$; $\mathcal{Q} \xleftarrow{\cup} (t,X,Y)$; $t \leftarrow t+1$;
3 return $X$;
```

Fig. 12. $\mathcal{Q}$ is a (initially empty) list and initially $t = 1$. In the step 1 of $(\mathcal{P}_1, \mathcal{P}_1^{-1})$, j is a maximum value.

We define a stateless simulator S in Fig. 11. Step 7 of S_F ensures the simulation of P and Step 7 of S_I ensures the simulation of P^{-1}. Steps 2-6 of S_F and Steps 2-6 of S_I ensure the simulation of the L-R relation.

Detail. In the following, for the simulator S in Fig. 11 and any distinguisher $\mathcal{D}$, we evaluate the bound of the reset indiferentiable advantage of FOLSponge^P from $\mathcal{WRO}$. To evaluate the bound we consider the following six games. In each game, $\mathcal{D}$ has oracle access to the left oracle L and the right oracles (R_F, R_I).

- Game 1 is the ideal world, that is, $(L, R_F, R_I) = (\mathcal{RO}_n, S_F, S_I)$.
- Game 2 is that $(\mathcal{P}, \mathcal{P}^{-1})$ are changed into $(\mathcal{P}_1, \mathcal{P}_1^{-1})$ shown in Fig. 12. So the simulator has oracle access to $(\mathcal{P}_1, \mathcal{P}_1^{-1})$ instead of $(\mathcal{P}, \mathcal{P}^{-1})$.
- Game 3 is $(L, R_F, R_I) = (\mathcal{RO}_n, S1_F, S1_I)$, where $S1 = (S1_F, S1_I)$ keeps all query-responses (X, Y) where $Y = S1_F(X)$ or $X = S1_I(Y)$. For a query X to $S1_F$, if there is (X, Y) in the query-response history, then $S1_F$ returns Y, otherwise, $S1_F$ returns the output of $S_F(X)$. For a query Y of $S1_I$, if there is (X, Y) in the query-response history, then $S1_I$ returns X, otherwise, $S1_I$ returns the output of $S_I(Y)$.
- Game 4 is $(L, R_F, R_I) = (L_1, S1_F, S1_I)$, where for a query M to L_1, first L_1 makes $S1_F$ queries corresponding with $\text{FOLSponge}^{S1_F}(M)$, and then returns the response of $\mathcal{RO}_n(M)$.
- Game 5 is $(L, R_F, R_I) = (\text{FOLSponge}^{S1_F}, S1_F, S1_I)$.
- Game 6 is the real world, that is, $(L, R_F, R_I) = (\text{FOLSponge}^P, P, P^{-1})$.

Let G_i be an event that $\mathcal{D}$ outputs 1 in Game i. We thus have that

$$\mathsf{Adv}^{\mathsf{r\text{-}indiff}}_{\text{FOLSponge}^P,\mathcal{WRO},S}(\mathcal{D}) \le \sum_{i=1}^{5} |\Pr[G_i] - \Pr[G_{i+1}]| \le \frac{2\sigma(\sigma+1) + q(q-1)}{2^c} + \frac{\sigma(\sigma-1) + q(q-1)}{2^{d+1}}.$$

In the following, we justify the above bound by evaluating each difference.

Game 1 ⇒ Game 2. From Game 1 to Game 2, we change the underlying oracle of (R_F, R_I) from $(\mathcal{P}, \mathcal{P}^{-1})$ to $(\mathcal{P}_1, \mathcal{P}_1^{-1})$ where $\mathcal{P}$ is a random permutation and $\mathcal{P}^{-1}$ is its inverse oracle, while responses of $\mathcal{P}_1$ and $\mathcal{P}_1^{-1}$ are uniformly chosen at random from $\{0,1\}^d$. Thus $|\Pr[G_1] - \Pr[G_2]|$ is bounded by the collision probability of $(\mathcal{P}_1, \mathcal{P}_1^{-1})$. Since $\mathcal{P}_1$ and $\mathcal{P}_1^{-1}$ are called at most q times,

$$|\Pr[G_1] - \Pr[G_2]| \leq \sum_{i=1}^{q} \frac{i-1}{2^d} = \frac{q(q-1)}{2^{d+1}}.$$

Game 2 ⇒ Game 3. From Game 2 to Game 3, we change (R_F, R_I) from (S_F, S_I) to $(S1_F, S1_I)$ where $(S1_F, S1_I)$ record query-response values while (S_F, S_I) don't record them. Therefore, if a query X to R_F (resp. Y to R_I) was made where the response is Y (resp. X), for a repeated query X to R_F (resp. Y to R_I) the same value Y (resp. X) is responded, while in Game 2 there is a case that for some repeated query $R_F(X)$ (resp. $R_I(Y)$) where Y (resp. X) was responded, a distinct value Y^* (resp. X^*) is responded. The difference $|\Pr[G_2] - \Pr[G_3]|$ is thus bounded by the probability that in Game 3 the distinct value is responded. We call the event "**Diff**". Since the procedures to define outputs of S_F and of S_I are controlled by $\mathcal{TO}$ (See the steps 2, 4, and 7 of S_F and the steps 2, 4, and 7 of S_I), the event **Diff** relies on responses of $\mathcal{TO}$. Therefore, if **Diff** occurs, for some repeated query to $\mathcal{TO}$, the response is changed. More precisely, if **Diff** occurs, the following event occurs.

- For a query y to $\mathcal{TO}$, w was responded. For the repeated query y to $\mathcal{TO}$, a distinct value w^* is responded. There are two cases for (w, w^*).
  - $\mathbf{Diff}_1$: $w = \perp$ and $w^* \neq \perp$.
  - $\mathbf{Diff}_2$: $w \neq \perp$ and $w^* = \perp$.

We thus have that

$$|\Pr[G_2] - \Pr[G_3]| \leq \Pr[\mathbf{Diff}_1] + \Pr[\mathbf{Diff}_2] \leq \frac{q(q-1)}{2^c}.$$

We justify the bound as follows.

First we bound the probability of $\Pr[\mathbf{Diff}_1]$. Since the response w of the first query is $\perp$, when the first query is made, the query w^* to $\mathcal{RO}^\dagger$ such that $y = \mathcal{RO}_c^\dagger(w^*)$ was not made. Since the response w^* of the repeated query is not $\perp$, when the repeated query is made, the query w^* to $\mathcal{RO}_c^\dagger$ was made such that $y = \mathcal{RO}_c^\dagger(w^*)$. Therefore, first y is defined. Second the response of $\mathcal{RO}_c^\dagger(w^*)$ hits y. Thus $\Pr[\mathbf{Diff}_1]$ is bounded by the probability that the response of $\mathcal{RO}_c^\dagger(w^*)$ (c-bit random value) hits the value y. Since the numbers of queries to $\mathcal{RO}_c^\dagger$ and $\mathcal{TO}$ are at most q times,

$$\Pr[\mathbf{Diff}_1] \leq \sum_{i=1}^{q} \frac{i-1}{2^c} \leq \frac{q(q-1)}{2^{c+1}}.$$

Next we bound the probability of $\Pr[\mathbf{Diff}_2]$. Since the response w of the first query is not $\perp$, when the first query is made, the query w to $\mathcal{RO}_c^\dagger$ was made such that $y = \mathcal{RO}_c^\dagger(w)$. Since the response w^* of the repeated query is $\perp$, when the repeated query is made, a query w' was made such that $w \neq w'$ and $\mathcal{RO}_c^\dagger(w) = \mathcal{RO}_c^\dagger(w')$. Therefore, $\Pr[\mathbf{Diff}_2]$ is bounded by the collision probability of $\mathcal{RO}_c^\dagger$. We thus have that

$$\Pr[\mathbf{Diff}_2] \leq \sum_{i=1}^{q} \frac{i-1}{2^c} \leq \frac{q(q-1)}{2^{c+1}}.$$

Game 3 ⇒ Game 4. From Game 3 to Game 4, we change L from $\mathcal{RO}_n$ to L_1 which makes additional right queries corresponding with FOLSponge^{S1_F}. Note that $\mathcal{D}$ cannot directly observe the additional right query-response values but can observe those by making the queries. So we must show that the additional right query-response values that $\mathcal{D}$ observes don't affect $\mathcal{D}$'s behavior. We ensure this by Lemma 2 where for any Sponge path $IV \xrightarrow{M} z$, $z = \mathcal{RO}_n(M) \| \mathcal{RO}_c^{\dagger}(M)$ unless Bad_j occurs. By Lemma 2, in both games, unless the bad event occurs, responses of queries to R_F are defined by the same queries to $\mathcal{RO}_c^{\dagger}$ and $\mathcal{RO}_n$. Namely, in Game 4, the responses of the additional queries to R_F which $\mathcal{D}$ observes are chosen from the same distribution as in Game 3 unless the bad event occurs. Thus, the difference $|\Pr[G_3] - \Pr[G_4]|$ is bounded by the probability of occurring the bad event.

First we define the bad event. Let T_i be a table which stores all values $X_t[n+1, d]$ and $Y_t[n+1, d]$ for $t = 1, \ldots, i-1$ where (X_t, Y_t) is a query-response pair defined by the t-th S_F or S_I query.

- Bad_j is that in Game j, for some i-th query X_i to S_F the response Y_i is such that $Y_i[n+1, d]$ collides with some value in $T_i \cup \{X_i[n+1, d]\} \cup \{IV_2\}$, or
  for some i-th query Y_i to S_I the response X_i is such that $X_i[n+1, d]$ collides with some value in $T_i \cup \{Y_i[n+1, d]\} \cup \{IV_2\}$.

Next we give Lemma 2. Lemma 2 is also used in the evaluation of the difference between Game 4 and Game 5.

Lemma 2. *In Game j, unless Bad_j occurs, for any Sponge path $IV \xrightarrow{M} z$ $z = \mathcal{RO}_n(M) \| \mathcal{RO}_c^{\dagger}(M)$.* ♦

Proof of Lemma 2. Assume that Bad_j does not occur. Let $IV \xrightarrow{M} z$ be any sponge path and $(X_1, Y_1), \ldots, (X_t, Y_t)$ be the corresponding pairs where $X_1[n+1, d] = IV_2$, $X_i[n+1, d] = Y_{i-1}[n+1, d]$ $(i = 2, \ldots, t)$, $Y_t = z$, and $M = M_1 || \ldots || M_t$ where $M_1 = IV_1 \oplus X_1[1, n], M_2 = Y_1[1, n] \oplus X_2[1, n], \cdots, M_t = Y_{t-1}[1, n] \oplus X_t[1, n]$. We show that $z = \mathcal{RO}_n(M) \| \mathcal{RO}_c^{\dagger}(M)$.

Consider the case of $t = 1$. Since Bad_j does not occur, there is no pair (X, Y) which is defined by an R_I query such that $X[n+1, d] = IV_2$. Thus any path $IV \xrightarrow{M} z$ is defined by an R_F query. Consequently, $z = \mathcal{RO}_n(M) \| \mathcal{RO}_c^{\dagger}(M)$ due to Steps 2 and 3 of S_F.

Consider the case of $t \geq 2$.

Since Bad_j does not occur, there is no pair (X_i, Y_i) which is defined by an R_I query and which connects with another pair (X_{i-1}, Y_{i-1}) which was already defined, that is, $(X_1, Y_1), \ldots, (X_t, Y_t)$ are defined by R_F queries. Since there is no pair (X_i, Y_i) which is defined by a R_F query and which connects with another pair (X_{i+1}, Y_{i+1}) which was already defined, $(X_1, Y_1), \ldots, (X_t, Y_t)$ are defined by the ordered R_F queries $S1_F(X_1), \ldots, S1_F(X_t)$.

Since no collision for $\mathcal{RO}_c^{\dagger}$ occurs, the structure of S_F ensures that when the query $R_F(X_t)$ is made, the pair $(M_1 || \cdots || M_{t-1}, Y_{t-1}[n+1, d])$ is stored in the table $\mathsf{F}^{\dagger}$, that is, $\mathsf{F}^{\dagger}[M_1 || \cdots || M_{t-1}] = Y_{t-1}[n+1, d]$.

Since no collision for $\mathcal{RO}_c^{\dagger}$ occurs, for the query $S_F(X_t)$, S_F makes the query $\mathcal{TO}(X_t[n+1, d])$, receives $M_1 || \cdots || M_{t-1}$ from $\mathcal{TO}$, and $S1_F$ returns the response of $\mathcal{RO}_n(M) \| \mathcal{RO}_c^{\dagger}(M)$. Thus $Y_t = \mathcal{RO}_n(M) \| \mathcal{RO}_c^{\dagger}(M)$.

□

By Lemma 2, we can bound the difference $|\Pr[G_3] - \Pr[G_4]|$ as follows.

$$|\Pr[G_3] - \Pr[G_4]| \leq \max\{\Pr[Bad_3], \Pr[Bad_4]\} \leq \frac{\sigma(\sigma+1)}{2^c}$$

where $\Pr[G_3 | \neg Bad_3] = \Pr[G_4 | \neg Bad_4]$ from Lemma 2.

Finally we justify the bound. The probability of occurring the bad event is that for some i-th query, the right c-bit value of the response of S_F, which is a random value, hits some of

$T_i \cup \{X_i[n+1,d]\} \cup \{IV_2\}$, or the right c-bit value of the response of S_I, which is a random value, hits some of $T_i \cup \{Y_i[n+1,d]\} \cup \{IV_2\}$. We thus have

$$\Pr[Bad_3] \leq \sum_{i=1}^{q} \frac{(2(i-1)+2)}{2^c} = \frac{q(q+1)}{2^c},\ \Pr[Bad_4] \leq \sum_{i=1}^{\sigma} \frac{(2(i-1)+2)}{2^c} = \frac{\sigma(\sigma+1)}{2^c}$$

where (S_F, S_I) are called at most q times in Game 3 and σ times in Game 4.

Game 4 ⇒ Game 5. From Game 4 to Game 5, we change L from L_1 to $\mathrm{FOLSponge}^{S_F}$. In Game 4 $L(M) = \mathcal{RO}_n(M)$, while in Game 5 $L(M) = \mathrm{FOLSponge}^{S_1}(M)$. Thus, the difference does not change $\mathcal{D}$'s behavior iff in Game 5 for any query M to L, L returns the response of $\mathcal{RO}_n(M)$. From Lemma 2, for any Sponge path $IV \xrightarrow{M} z$ the relation $z[1,n] = \mathcal{RO}_n(M)$ holds unless the bad event Bad_5 occurs. Therefore, the difference $|\Pr[G_4] - \Pr[G_5]|$ is bounded by the probability of occurring the bad event Bad_5. In Game 5 R is called at most σ times and for any query to S the response is chosen uniformly at random from $\{0,1\}^c$. We have that

$$|\Pr[G_4] - \Pr[G_5]| \leq \Pr[Bad_5] \leq \frac{\sigma(\sigma+1)}{2^c}.$$

Game 5 ⇒ Game 6. From Game 5 to Game 6, we change (R_F, R_I) from $(S1_F, S1_I)$ to (P, P^{-1}), where outputs of $(S1_F, S1_I)$ are chosen uniformly at random from $\{0,1\}^d$, while (P, P^{-1}) are a random permutation and its inverse oracle. The difference is thus bounded by the collision probability of outputs of $(S1_F, S1_I)$ in Game 5. We thus have that

$$|\Pr[G_5] - \Pr[G_6]| \leq \sum_{i=1}^{\sigma} \frac{i-1}{2^d} = \frac{\sigma(\sigma-1)}{2^{d+1}}.$$

□

C Proof of Theorem 4

Proof. We denote $\mathsf{Adv}(\mathcal{A}, \mathbf{G}_i)$ by the advantage of the adversary $\mathcal{A}$ when participating in experiment $\mathbf{G}_i$. We start with game $\mathbf{G}_0$ which is exactly the same game as the CDA game in the $\mathcal{WRO}$ model. It means $\mathsf{Adv}(\mathcal{A}, \mathbf{G}_0) = \mathsf{Adv}^{\mathrm{cda}}_{\mathcal{AE},\mathcal{WRO}}(\mathcal{A}_1, \mathcal{A}_2)$.

Game $\mathbf{G}_1$: $\mathcal{RO}_n$ returns a random value if the following event occurs:

- Bad_1 : $\mathcal{A}_1$ poses a message M to $\mathcal{RO}_n$ where M is posed to $\mathcal{RO}_n$ by Enc to generate the challenge ciphertext.

All other procedures are computed as the same way in $\mathbf{G}_0$.

Lemma 3. $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_1) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_0)| \leq q_{\mathcal{RO}} \cdot \mathsf{maxpk}_{\mathcal{AE}}$.

Proof. The difference between $\mathbf{G}_0$ and $\mathbf{G}_1$ only occurs in Bad_1. From Difference Lemma [22], we have that $|\mathsf{Adv}(\mathcal{B}, \mathbf{G}_1) - \mathsf{Adv}(\mathcal{B}, \mathbf{G}_0)| \leq \Pr[\mathsf{Bad}_1]$.

We estimate $\Pr[\mathsf{Bad}_1]$. Since pk is not given for $\mathcal{A}_1$ and is included in each query to $\mathcal{RO}_n$ by Enc, the only way to pose $(pk, *, *)$ to $\mathcal{RO}_n$ is choosing pk randomly $q_{\mathcal{RO}}$ times. We have that $\Pr[\mathsf{Bad}_1] \leq q_{\mathcal{RO}} \cdot \mathsf{maxpk}_{\mathcal{AE}}$. □

```
Game $\mathbf{G}_2$
$\beta \xleftarrow{\$} \{0,1\}$
$(pk, sk) \xleftarrow{\$} \mathsf{Gen}$
$(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r}) \leftarrow \mathcal{A}_1^{\mathcal{RO}_n, \mathcal{RO}_v^*, \mathcal{RO}_w^\dagger, \mathcal{TO}, \mathsf{IC}_{a,b}}$
$\mathbf{c} \leftarrow \mathsf{Enc}^{F.hon}(pk, \mathbf{m}_\beta; \mathbf{r})$
$\mathbf{c}' \leftarrow \mathcal{S}^{\mathcal{RO}_n}(pk, \omega)$
$\beta' \leftarrow \mathcal{A}_2^{\mathcal{RO}_n, \mathcal{RO}_v^*, \mathcal{RO}_w^\dagger, \mathcal{TO}, \mathsf{IC}_{a,b}}(pk, \mathbf{c}')$
return $(\beta = \beta')$
```

```
$\mathcal{B}^{\mathsf{RoS}}(pk)$
$\beta \xleftarrow{\$} \{0,1\}$
$(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r}) \leftarrow \mathcal{A}_1^{\mathsf{SimB}}$
$\mathbf{c} \leftarrow \mathsf{RoS}(\mathbf{m}_\beta, \mathbf{r})$
$\beta' \leftarrow \mathcal{A}_2^{\mathsf{SimB}}(pk, \mathbf{c})$
If $\beta = \beta'$ then return 1
Otherwise return 0
```

```
$\mathsf{SimB}_{\mathcal{RO}_n}(M)$
If $\mathsf{F}[M] =\perp$, $\mathsf{F}[M] \xleftarrow{\$} \{0,1\}^n$
If $\mathsf{F}[M] \neq\perp$, and $M$ is posed by $\mathsf{Enc}$,
    $\mathsf{F}[M] \xleftarrow{\$} \{0,1\}^n$
return $\mathsf{F}[M]$
```

```
$\mathsf{SimB}_{\mathcal{RO}_v^*}(M)$
If $\mathsf{F}^*[M] =\perp$, $\mathsf{F}^*[M] \xleftarrow{\$} \{0,1\}^v$
return $\mathsf{F}^*[M]$;
```

```
$\mathsf{SimB}_{\mathcal{RO}_w^\dagger}(M)$
If $\mathsf{F}^\dagger[M] =\perp$ then $\mathsf{F}^\dagger[M] \xleftarrow{\$} \{0,1\}^w$
return $\mathsf{F}^\dagger[M]$;
```

```
$\mathsf{SimB}_{\mathcal{TO}}(y)$
If $\exists_1 M$ s.t. $\mathsf{F}^\dagger[M] = y$ then return $M$
Otherwise return $\perp$
```

```
$\mathsf{SimB}_E(k, x)$
If $\mathsf{E}[k, x] =\perp$,
    $y \xleftarrow{\$} \{0,1\}^b \backslash T^+[k]$
    $\mathsf{E}[k, x] \leftarrow y, \mathsf{D}[k, y] \leftarrow x$,
    $T^+[k] \xleftarrow{\cup} \{y\}$, $T^-[k] \xleftarrow{\cup} \{x\}$
return $\mathsf{E}[k, x]$
```

```
$\mathsf{SimB}_D(k, y)$
If $\mathsf{D}[k, y] =\perp$,
    $x \xleftarrow{\$} \{0,1\}^b \backslash T^-[k]$;
    $\mathsf{E}[k, x] \leftarrow y, \mathsf{D}[k, y] \leftarrow x$,
    $T^+[k] \xleftarrow{\cup} \{y\}$, $T^-[k] \xleftarrow{\cup} \{x\}$
return $\mathsf{D}[k, y]$
```

Fig. 13. game $\mathbf{G}_2$ and simulation SimB by adversary $\mathcal{B}$

Game $\mathbf{G}_2$: Ciphertext $\mathbf{c} \leftarrow \mathsf{Enc}^{\mathcal{RO}_n}(pk, \mathbf{m}_b; \mathbf{r})$ is replaced with outputs of a simulator $\mathcal{S}^{\mathcal{RO}_n}(pk, \omega)$ in the IND-SIM game. All other procedures are computed as the same way in $\mathbf{G}_1$.

Lemma 4. $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_2) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_1)| \leq \mathsf{Adv}^{\text{ind-sim}}_{\mathcal{AE}, \mathcal{S}, \mathcal{RO}_n}(\mathcal{B})$.

Proof. We show that if $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_2) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_1)|$ is non-negligible, for any simulator $\mathcal{S}$ we can construct an adversary $\mathcal{B}$ breaking IND-SIM security of $\mathcal{AE}$ in the RO model. Fig. 13 shows game $\mathbf{G}_2$, the construction of $\mathcal{B}$, and the simulation $\mathsf{SimB} = (\mathsf{SimB}_{\mathcal{RO}_n}, \mathsf{SimB}_{\mathcal{RO}_v^*}, \mathsf{SimB}_{\mathcal{RO}_w^\dagger}, \mathsf{SimB}_{\mathcal{TO}}, \mathsf{SimB}_E, \mathsf{SimB}_D)$ of $\mathcal{WRO}$ by $\mathcal{B}$ respectively. Note that $\mathcal{B}$ makes no RO queries, and $\mathsf{Enc}^{F.hon}(pk, \mathbf{m}_\beta; \mathbf{r})$ is executed with return value ignored. $\mathcal{B}$ simulates all queries to $\mathcal{WRO}$ for $\mathcal{A}_1$ and $\mathcal{A}_2$ with simulation SimB. SimB is identical with the definition of $\mathcal{WRO}$. Also, queries to $\mathcal{RO}_n$ by Enc is contained both in $\mathbf{G}_1$ and $\mathbf{G}_2$. Thus, $\mathcal{A}$ cannot distinguish game $\mathbf{G}_1$ and $\mathbf{G}_2$ from the simulation on the interface of $\mathcal{WRO}$. If $\beta = 1$ in IND-SIM game, it is clear that all interfaces for $\mathcal{A}$ is exactly same as game $\mathbf{G}_1$. If $\beta = 0$ in IND-SIM game, it is clear that all interfaces for $\mathcal{A}$ is exactly same as game $\mathbf{G}_2$.

Therefore, if $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_2) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_1)|$ is non-negligible, $\mathcal{B}$ also breaks IND-SIM security of $\mathcal{AE}$. □

Game $\mathbf{G}_3$: When $\mathcal{A}_2$ poses a query related to $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$ (which is the output of $\mathcal{A}_1$) to $\mathcal{RO}_v^*, \mathcal{RO}_w^\dagger, \mathcal{TO}$ or $\mathsf{IC}_{a,b} = (E, D)$, then outputs are randomly chosen. That is, tables $\mathsf{F}^*, \mathsf{F}^\dagger, \mathsf{E}$ and D are not preserved for $\mathcal{A}_1$ and $\mathcal{A}_2$ according to $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$. All other procedures are computed as the same way in $\mathbf{G}_2$.

Lemma 5. $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_3) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_2)| \leq \frac{4q^2_{\mathcal{RO}^*} + 4q^2_{\mathcal{RO}^\dagger}}{2^\mu} + max\left\{\frac{4q^2_{\mathcal{TO}}}{2^\mu}, \frac{4q^2_{\mathcal{TO}}}{2^w}\right\} + max\left\{\frac{4q^2_E + 4q^2_D}{2^\mu}, \frac{4q^2_E + 4q^2_D}{2^b}\right\}$.

Proof. The difference between $\mathbf{G}_2$ and $\mathbf{G}_3$ only occurs when $\mathcal{A}_1$ and $\mathcal{A}_2$ poses a same query related to $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$ to $\mathcal{RO}_v^*, \mathcal{RO}_w^\dagger, \mathcal{TO}$ or $\mathsf{IC}_{a,b} = (E, D)$. We denote the event that a common query related to $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$ is posed to $\mathcal{RO}_v^*$ by $\mathcal{A}_1$ and $\mathcal{A}_2$ as $\mathsf{Bad}_{\mathcal{RO}^*}$. Similarly, we define events $\mathsf{Bad}_{\mathcal{RO}^\dagger}$,

$\mathsf{Bad}_{\mathcal{TO}}$, Bad_E, and Bad_D. From Difference Lemma [22], we have that $|\mathsf{Adv}(\mathcal{B}, \mathbf{G}_3) - \mathsf{Adv}(\mathcal{B}, \mathbf{G}_2)| \leq \Pr[\mathsf{Bad}_{\mathcal{RO}^*} \vee \mathsf{Bad}_{\mathcal{RO}^\dagger} \vee \mathsf{Bad}_{\mathcal{TO}} \vee \mathsf{Bad}_E \vee \mathsf{Bad}_D] \leq \Pr[\mathsf{Bad}_{\mathcal{RO}^*}] + \Pr[\mathsf{Bad}_{\mathcal{RO}^\dagger}] + \Pr[\mathsf{Bad}_{\mathcal{TO}}] + \Pr[\mathsf{Bad}_E] + \Pr[\mathsf{Bad}_D]$.

In game $\mathbf{G}_2$ and $\mathbf{G}_3$, ciphertext $\mathbf{c}$ does not give any information about $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$ and queries to $\mathcal{WRO}$ by $\mathcal{A}_1$ to $\mathcal{A}_2$. On queries to $\mathcal{RO}_n$, interfaces of $\mathcal{A}_2$ in $\mathbf{G}_2$ and $\mathbf{G}_3$ are identical. Thus, the only way to pose such a query is guessing under min-entropy μ, or the output length w of $\mathcal{RO}_T$ and the output length b of (E, D). According to the birthday paradox, for oracles $\mathcal{RO}^*$ and $\mathcal{RO}^\dagger$ the probability of collisions in guessing is at most $(2q_{\mathcal{RO}^*})^2/2^\mu$, and $(2q_{\mathcal{RO}^\dagger})^2/2^\mu$, respectively. Also, for oracle $\mathcal{TO}$ the probability of collision in guessing is at most $(2q_{\mathcal{TO}})^2/2^\mu$ if $\mu < w$, $(2q_{\mathcal{TO}})^2/2^w$ otherwise, and for oracles E and D the probability of collisions in guessing is at most $(2q_E)^2/2^\mu$ and $(2q_D)^2/2^\mu$ if $\mu < b$, $(2q_E)^2/2^b$ and $(2q_D)^2/2^b$ otherwise. Therefore, $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_3) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_2)| \leq (4q^2_{\mathcal{RO}^*} + 4q^2_{\mathcal{RO}^\dagger})/2^\mu + max\left\{4q^2_{\mathcal{TO}}/2^\mu, 4q^2_{\mathcal{TO}}/2^w\right\} + max\left\{(4q^2_E + 4q^2_D)/2^\mu, (4q^2_E + 4q^2_D)/2^b\right\}$. □

Game $\mathbf{G}_4$: $\mathcal{RO}_n$ returns a random value if the following event occurs:

- Bad_2 : $\mathcal{A}_2$ poses a message M to $\mathcal{RO}_n$ where M is posed to $\mathcal{RO}_n$ by Enc to generate the challenge ciphertext.

All other procedures are computed as the same way in $\mathbf{G}_3$.

Lemma 6. $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_4) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_3)| \leq \frac{q_{\mathcal{RO}}}{2^\mu}$.

Proof. The difference between $\mathbf{G}_3$ and $\mathbf{G}_4$ only occurs in Bad_2. From Difference Lemma [22], we have that $|\mathsf{Adv}(\mathcal{B}, \mathbf{G}_4) - \mathsf{Adv}(\mathcal{B}, \mathbf{G}_3)| \leq \Pr[\mathsf{Bad}_2]$.

We estimate $\Pr[\mathsf{Bad}_2]$. Because $\mathcal{RO}_n$ is a truly random function, ciphertext $\mathbf{c}$ is replaced, and $\mathcal{A}_1$ does not pose related queries to M to sub-oracles, $\mathcal{A}_2$ cannot obtain more information of r than min-entropy μ. Thus, the only way to pose M to $\mathcal{RO}_n$ is guessing M under min-entropy μ $q_{\mathcal{RO}}$ times. We have that $\Pr[\mathsf{Bad}_2] \leq \frac{q_{\mathcal{RO}}}{2^\mu}$. □

We estimate $\mathsf{Adv}(\mathcal{A}, \mathbf{G}_4)$. Ciphertext $\mathbf{c}$ does not give any information about $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$. Also, outputs of $\mathcal{WRO}$ is independent of $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$ for $\mathcal{A}_2$. Thus, the only way to win in game $\mathbf{G}_4$ is randomly guessing β. Therefore, $\mathsf{Adv}(\mathcal{A}, \mathbf{G}_4) = 0$.

To conclude, we have $\mathsf{Adv}^{\text{cda}}_{\mathcal{AE},\mathcal{WRO}}(\mathcal{A}_1, \mathcal{A}_2) \leq \mathsf{Adv}^{\text{ind-sim}}_{\mathcal{AE},\mathcal{S},\mathcal{RO}_n}(\mathcal{B}) + q_{\mathcal{RO}} \cdot \mathsf{maxpk}_{\mathcal{AE}} + (q_{RO} + 4q^2_{\mathcal{RO}^*} + 4q^2_{\mathcal{RO}^\dagger})/2^\mu + max\left\{4q^2_{\mathcal{TO}}/2^\mu, 4q^2_{\mathcal{TO}}/2^w\right\} + max\left\{(4q^2_E + 4q^2_D)/2^\mu, (4q^2_E + 4q^2_D)/2^b\right\}$. □

D ID-CDA Secure IBE in the $\mathcal{WRO}$ Model

ID-based Encryption (IBE). An ID-based encryption scheme $\mathcal{IBE} = (\mathsf{IBE.Setup}, \mathsf{IBE.Gen}, \mathsf{IBE.Enc}, \mathsf{IBEDec})$ consists of four algorithms. Setup algorithm $\mathsf{IBE.Setup}$ outputs public parameter $params$ and master secret key msk. Key generation algorithm $\mathsf{IBE.Gen}$ takes public parameter $params$, master secrete key msk and ID id, and outputs secret key sk for id. Encryption algorithm $\mathsf{IBE.Enc}$ takes public parameter $params$, ID id, plaintext m, and randomness r, and outputs ciphertext c. Decryption algorithm IBEDec takes public parameter $params$, secret key sk, and ciphertext c, and outputs plaintext m or distinguished symbol $\perp$. For vectors $\mathbf{m}, \mathbf{r}$ with $|\mathbf{m}| = |\mathbf{r}| = l$ which is the size of vectors, we denote by $\mathsf{IBE.Enc}(params, id, \mathbf{m}; \mathbf{r})$ the vector $(\mathsf{IBE.Enc}(params, id, \mathbf{m}[1]; \mathbf{r}[1]), \ldots, \mathsf{IBE.Enc}(params, id, \mathbf{m}[l]; \mathbf{r}[l]))$. We say that $\mathcal{IBE}$ is deterministic if $\mathsf{IBE.Enc}$ is deterministic.

```
$\underline{\text{ID-CPA}^{\mathcal{B}}_{\mathcal{IBE},F}}$
$\beta \xleftarrow{\$} \{0,1\}$
If selective-ID setting
    $id^* \leftarrow \mathcal{B}^{F.adv,\mathsf{IBE.Gen}(\cdot)}$
$(params, msk) \xleftarrow{\$} \mathsf{IBE.Setup}$
If selective-ID setting
    $(\mathbf{m}_0, \mathbf{m}_1) \leftarrow \mathcal{B}^{F.adv,\mathsf{IBE.Gen}(\cdot)}(params)$
If full-ID setting
    $(\mathbf{m}_0, \mathbf{m}_1, id^*) \leftarrow \mathcal{B}^{F.adv,\mathsf{IBE.Gen}(\cdot)}(params)$
$\mathbf{r} \xleftarrow{\$} (\{0,1\}^{\rho})^l$
$\mathbf{c} \leftarrow \mathsf{IBE.Enc}^{F.hon}(params, id^*, \mathbf{m}_\beta; \mathbf{r})$
$\beta' \leftarrow \mathcal{B}^{F.adv,\mathsf{IBE.Gen}(\cdot)}(\mathbf{c})$
return $(\beta = \beta')$
```

```
$\underline{\text{ID-CDA}^{\mathcal{A}_1,\mathcal{A}_2}_{\mathcal{IBE},F}}$
$\beta \xleftarrow{\$} \{0,1\}$
$id^* \leftarrow \mathcal{A}_2^{F.adv,\mathsf{IBE.Gen}(\cdot)}$
$(params, msk) \xleftarrow{\$} \mathsf{IBE.Setup}$
$(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r}) \leftarrow \mathcal{A}_1^{F.adv,\mathsf{IBE.Gen}(\cdot)}$
$\mathbf{c} \leftarrow \mathsf{IBE.Enc}^{F.hon}(params, id^*, \mathbf{m}_\beta; \mathbf{r})$
$\beta' \leftarrow \mathcal{A}_2^{F.adv,\mathsf{IBE.Gen}(\cdot)}(params, \mathbf{c})$
return $(\beta = \beta')$
```

Fig. 14. ID-CPA and ID-CDA game

ID-based CPA and CDA Security. We define the ID-CPA and the (non-adaptive) ID-CDA security. The ID-CPA security is a standard one [9–11] except that an adversary can pose multiple challenge plaintext pairs. It is known that the CPA game with multiple challenge is polynomial-time reducible to the game with single challenge. Let $\mathcal{CH}$ be the challenger of the ID-CPA game. The ID-CDA security is based on the CDA security. Fig. 14 illustrates the ID-CPA game and the non-adaptive ID-CDA game in the CPA case for $\mathcal{IBE}$ using a functionality F. In both games, $\mathcal{B}$ and $\mathcal{A}$ are not allowed to pose id^* to $\mathsf{IBE.Gen}(\cdot)$. As the CDA security, the ID-CDA adversary $\mathcal{A}_1$ is a (μ, ν)-mmr-source. (1) The advantage for an ID-CPA adversary $\mathcal{B}$ against scheme $\mathcal{IBE}$ using a functionality F and (2) the advantage for an ID-CDA adversary $(\mathcal{A}_1, \mathcal{A}_2)$ against scheme $\mathcal{IBE}$ using a functionality F are defined by

$$(1)\ \mathsf{Adv}^{\text{id-cpa}}_{\mathcal{IBE},F}(\mathcal{B}) = 2 \cdot \Pr[\text{ID-CPA}^{\mathcal{B}}_{\mathcal{IBE},F} \Rightarrow \mathsf{true}] - 1.$$

$$(2)\ \mathsf{Adv}^{\text{id-cda}}_{\mathcal{IBE},F}(\mathcal{A}_1, \mathcal{A}_2) = 2 \cdot \Pr[\text{ID-CDA}^{\mathcal{A}_1,\mathcal{A}_2}_{\mathcal{IBE},F} \Rightarrow \mathsf{true}] - 1.$$

Hedged ID-based Encryption IDREwH1. We show an example of ID-CDA secure hedged IBE, IDREwH1. The proposed scheme is a simple extension of REwH1 [2].

Let $\mathcal{IBE}_r = (\mathsf{IBE.Setup}_r, \mathsf{IBE.Gen}_r, \mathsf{IBE.Enc}_r, \mathsf{IBEDec}_r)$ be an IBE scheme with plaintext length λ and randomness length ρ. $\mathcal{RO}_n$ has range size $\rho = n$ bits. $\mathsf{IDREwH1} = (\mathsf{IBE.Setup}_r, \mathsf{IBE.Gen}_r, \mathsf{IBE.Enc}, \mathsf{IBEDec}_r)$ uses same algorithms as $\mathcal{IBE}_r$ except $\mathsf{IBE.Enc}$ which is defined as

$$\mathsf{IBE.Enc}^{\mathcal{RO}_n}(params, id, m; r) = \mathsf{IBE.Enc}_r(params, id, m; \mathcal{RO}_n(params, id, m, r)).$$

If $|\rho| = 0$, we can obtain an ID-based version of a deterministic encryption scheme, Encrypt-with-Hash. Our theorems about IDREwH1 also work for deterministic encryption.

ID-CPA Security of IDREwH1 in the $\mathcal{WRO}$ Model. We can prove the ID-CPA security of IDREwH1; that is, we show that IDREwH1 is selective (resp. full) ID-CPA secure in the $\mathcal{WRO}$ model if $\mathcal{IBE}_r$ is selective (resp. full) ID-CPA secure in the RO model,

Theorem 5. *Let $\mathcal{IBE}_r$ be an IBE scheme. Let $\mathcal{B}$ be a selective (resp. full) CPA adversary for* IDREwH1 *in the $\mathcal{WRO}$ model, which makes at most $q_{\mathcal{RO}}, q_{\mathcal{RO}^*}, q_{\mathcal{RO}^\dagger}, q_{\mathcal{TO}}, q_E, q_D$ queries to $\mathcal{RO}_n, \mathcal{RO}^*_v$,*

$\mathcal{RO}^{\dagger}_{w}, \mathcal{TO}, \mathsf{IC}_{a,b} = (E, D)$. Then, there exists a selective (resp. full) CPA adversary $\mathcal{C}$ for $\mathcal{IBE}_r$ such that

$$\mathsf{Adv}^{\text{id-cpa}}_{\mathsf{IDREwH1},\mathcal{WRO}}(\mathcal{B}) \leq \mathsf{Adv}^{\text{id-cpa}}_{\mathcal{IBE}_r,\mathcal{RO}}(\mathcal{C}) + \frac{q_{\mathcal{RO}}}{2^{\rho}}.$$

$\mathcal{C}$ runs in time that of $\mathcal{B}$ plus $\mathcal{O}(q_{\mathcal{RO}} + q_{\mathcal{RO}^*} + q_{\mathcal{RO}^{\dagger}} + q_{\mathcal{TO}} + q_E + q_D)$. ♦

The proof outline is as follows: First, we start with game $\mathbf{G}_0$ which is exactly the same game as the ID-CPA game in the $\mathcal{WRO}$ model. Next, we transform $\mathbf{G}_0$ to game $\mathbf{G}_1$ so that challenge ciphertext $\mathbf{c}$ is generated from fresh randomness instead of the output of $\mathcal{RO}_n$. In game $\mathbf{G}_1$, $\mathbf{c}$ is generated by the exactly same manner as the ID-CPA game for $\mathcal{IBE}_r$. Also, oracle queries to $\mathcal{WRO}$ except $\mathcal{RO}_n$ is perfectly simulated because $\mathsf{IBE.Enc}$ algorithm never use $\mathcal{RO}^*_v, \mathcal{RO}^{\dagger}_w, \mathcal{TO}, E, D$. Thus, $\mathcal{B}$ can be constructed with $\mathcal{C}$.

Proof. We denote $\mathsf{Adv}(\mathcal{B}, \mathbf{G}_i)$ by the advantage of adversary $\mathcal{B}$ when participating in experiment $\mathbf{G}_i$. We start with game $\mathbf{G}_0$ which is exactly the same game as the ID-CPA game in the $\mathcal{WRO}$ model. It means $\mathsf{Adv}(\mathcal{B}, \mathbf{G}_0) = \mathsf{Adv}^{\text{id-cpa}}_{\mathsf{IDREwH1},\mathcal{WRO}}(\mathcal{B})$.

Game $\mathbf{G}_1$: Challenge ciphertext $\mathbf{c} \leftarrow \mathsf{IBE.Enc}_r(params, id^*, \mathbf{m}_\beta; \mathcal{RO}(params, id^*, \mathbf{m}_\beta; \mathbf{r}))$ is replaced with $\mathbf{c} \leftarrow \mathsf{IBE.Enc}_r(params, id^*, \mathbf{m}_\beta; \mathbf{r}')$ for randomly chosen $\mathbf{r}'$. All other procedures are computed as the same way in $\mathbf{G}_0$.

Lemma 7. $|\mathsf{Adv}(\mathcal{B}, \mathbf{G}_1) - \mathsf{Adv}(\mathcal{B}, \mathbf{G}_0)| \leq \frac{q_{\mathcal{RO}}}{2^{\rho}}$.

Proof. The difference between $\mathbf{G}_0$ and $\mathbf{G}_1$ only occurs when adversary $\mathcal{B}$ poses $(params, id^*, m_\beta, r)$ to $\mathcal{RO}_n$ where $m_\beta \in \mathbf{m}_\beta$, $r \in \mathbf{r}$, and $\mathbf{r}$ is the randomness vector used to generate challenge ciphertext $\mathbf{c}$. We denote this event as Bad. From Difference Lemma [22], we have that $|\mathsf{Adv}(\mathcal{B}, \mathbf{G}_1) - \mathsf{Adv}(\mathcal{B}, \mathbf{G}_0)| \leq \Pr[\mathsf{Bad}]$.

We estimate $\Pr[\mathsf{Bad}]$. Since $\mathcal{RO}_n$ is a truly random function, $\mathcal{B}$ cannot know $\mathbf{r}$ (which is used to generate challenge ciphertext $\mathbf{c}$) from challenge ciphertext even if $\mathcal{B}$ could obtain some information about $\mathcal{RO}_n(params, id^*, \mathbf{m}_\beta; \mathbf{r})$ from $\mathbf{c}$. Thus, the only way to pose $(params, id^*, m_\beta, r)$ to $\mathcal{RO}_n$ is choosing r randomly $q_{\mathcal{RO}}$ times. We have that $\Pr[\mathsf{Bad}] \leq \frac{q_{\mathcal{RO}}}{2^{\rho}}$. □

We estimate $\mathsf{Adv}(\mathcal{B}, \mathbf{G}_1)$. We assume that there exists $\mathcal{B}$ with $\mathsf{Adv}(\mathcal{B}, \mathbf{G}_1)$. Then, we construct adversary $\mathcal{C}$ against $\mathcal{IBE}_r$ with the same advantage as $\mathsf{Adv}(\mathcal{B}, \mathbf{G}_1)$. The simulation SimC by $\mathcal{C}$ is given in Fig. 15.

Since the generation of the challenge ciphertext is exactly same between $\mathbf{G}_0$ and the ID-CPA game for $\mathcal{IBE}_r$, $\mathcal{C}$ just forwards the challenge ciphertext to $\mathcal{B}$. The simulation of $\mathcal{WRO}$ is perfect because the challenger $\mathcal{CH}$ never uses all components of $\mathcal{WRO}$ with the honest interface. Therefore, $\mathsf{Adv}(\mathcal{B}, \mathbf{G}_1) = \mathsf{Adv}^{\text{id-cpa}}_{\mathcal{IBE}_r,\mathcal{RO}}(\mathcal{C})$.

To conclude, we have $\mathsf{Adv}^{\text{id-cpa}}_{\mathsf{IDREwH1},\mathcal{WRO}}(\mathcal{B}) \leq \mathsf{Adv}^{\text{id-cpa}}_{\mathcal{IBE}_r,\mathcal{RO}}(\mathcal{C}) + \frac{q_{\mathcal{RO}}}{2^{\rho}}$. □

ID-CDA Security of $\mathsf{IDREwH1}$ in the $\mathcal{WRO}$ Model. We prove the ID-CDA security of $\mathsf{IDREwH1}$; that is, we show that $\mathsf{IDREwH1}$ is ID-CDA secure in the $\mathcal{WRO}$ model if $\mathcal{IBE}_r$ is selective ID-CPA secure in the RO model.

Theorem 6. *Let $\mathcal{IBE}_r$ be an IBE scheme. Let $(\mathcal{A}_1, \mathcal{A}_2)$ be a CDA adversary for $\mathsf{IDREwH1}$ in the $\mathcal{WRO}$ model, which makes at most $q_{\mathcal{RO}}, q_{\mathcal{RO}^*}, q_{\mathcal{RO}^{\dagger}}, q_{\mathcal{TO}}, q_E, q_D$ queries to $\mathcal{RO}_n, \mathcal{RO}^*_v, \mathcal{RO}^{\dagger}_w, \mathcal{TO}, \mathsf{IC}_{a,b} =$*

```
$\underline{\mathsf{SimC}_{\mathrm{main}}}$
If selective-ID setting
    receive $id^*$ from $\mathcal{B}$
    send $id^*$ to $\mathcal{CH}$
receive $params$ from $\mathcal{CH}$
send $params$ to $\mathcal{B}$
If selective-ID setting
    $(\mathbf{m}_0, \mathbf{m}_1) \leftarrow \mathcal{B}$
    send $(\mathbf{m}_0, \mathbf{m}_1)$ to $\mathcal{CH}$
If full-ID setting
    $(\mathbf{m}_0, \mathbf{m}_1, id^*) \leftarrow \mathcal{B}$
    send $(\mathbf{m}_0, \mathbf{m}_1, id^*)$ to $\mathcal{CH}$
receive $\mathbf{c}$ from $\mathcal{CH}$
send $\mathbf{c}$ to $\mathcal{B}$
receive $\beta'$ from $\mathcal{B}$
return $\beta'$
```

```
$\underline{\mathsf{SimC}_{\mathsf{IBE.Gen}}(id)}$
send $id$ to IBE.Gen oracle
receive $sk_{id}$ from IBE.Gen oracle
return $sk_{id}$

$\underline{\mathsf{SimC}_{\mathcal{RO}_n}(M)}$
send $M$ to $\mathcal{RO}$
receive $\mathsf{F}[M]$ from $\mathcal{RO}$
return $\mathsf{F}[M]$

$\underline{\mathsf{SimC}_{\mathcal{RO}^*_v}(M)}$
If $\mathsf{F}^*[M] = \perp$, $\mathsf{F}^*[M] \xleftarrow{\$} \{0,1\}^v$
return $\mathsf{F}^*[M]$;

$\underline{\mathsf{SimC}_{\mathcal{RO}^\dagger_w}(M)}$
If $\mathsf{F}^\dagger[M] = \perp$ then $\mathsf{F}^\dagger[M] \xleftarrow{\$} \{0,1\}^w$
return $\mathsf{F}^\dagger[M]$;
```

```
$\underline{\mathsf{SimC}_{\mathcal{TO}}(y)}$
If $\exists_1 M$ s.t. $\mathsf{F}^\dagger[M] = y$ then return $M$
Otherwise return $\perp$

$\underline{\mathsf{SimC}_E(k,x)}$
If $\mathsf{E}[k,x] = \perp$,
    $y \xleftarrow{\$} \{0,1\}^b \backslash T^+[k]$
    $\mathsf{E}[k,x] \leftarrow y, \mathsf{D}[k,y] \leftarrow x$,
    $T^+[k] \xleftarrow{\cup} \{y\}$, $T^-[k] \xleftarrow{\cup} \{x\}$
return $\mathsf{E}[k,x]$

$\underline{\mathsf{SimC}_D(k,y)}$
If $\mathsf{D}[k,y] = \perp$,
    $x \xleftarrow{\$} \{0,1\}^b \backslash T^-[k]$;
    $\mathsf{E}[k,x] \leftarrow y, \mathsf{D}[k,y] \leftarrow x$,
    $T^+[k] \xleftarrow{\cup} \{y\}$, $T^-[k] \xleftarrow{\cup} \{x\}$
return $\mathsf{D}[k,y]$
```

Fig. 15. Simulation SimC by adversary $\mathcal{C}$

(E, D). Then, there exists a selective CPA adversary $\mathcal{C}$ for $\mathcal{IBE}_r$ such that

$$\mathsf{Adv}^{\text{id-cda}}_{\mathsf{IDREwH1},\mathcal{WRO}}(\mathcal{A}_1, \mathcal{A}_2) \leq 2\mathsf{Adv}^{\text{id-cpa}}_{\mathcal{IBE}_r,\mathcal{RO}}(\mathcal{C}) + q_{\mathcal{RO}} \cdot \mathsf{maxparams}_{\mathcal{IBE}_r} + \frac{q_{\mathcal{RO}} + 4q^2_{\mathcal{RO}^*} + 4q^2_{\mathcal{RO}^\dagger}}{2^\mu}$$
$$+ max\left\{\frac{4q^2_{\mathcal{TO}}}{2^\mu}, \frac{4q^2_{\mathcal{TO}}}{2^w}\right\} + max\left\{\frac{4q^2_E + 4q^2_D}{2^\mu}, \frac{4q^2_E + 4q^2_D}{2^b}\right\}.$$

$\mathcal{C}$ runs in time that of $(\mathcal{A}_1, \mathcal{A}_2)$ plus $\mathcal{O}(q_{\mathcal{RO}} + q_{\mathcal{RO}^*} + q_{\mathcal{RO}^\dagger} + q_{\mathcal{TO}} + q_E + q_D)$. $\mathsf{maxparams}_{\mathcal{IBE}_r}$ is the maximum public-parameter collision probability defined as $\mathsf{maxparams}_{\mathcal{IBE}_r} = \max_{\gamma \in \{0,1\}^*} \Pr[params = \gamma : (params, msk) \xleftarrow{\$} \mathsf{IBE.Setup}]$. ♦

The proof outline is as follows: First, we start with game $\mathbf{G}_0$ which is exactly the same game as the ID-CDA game in the $\mathcal{WRO}$ model. Secondly, we transform $\mathbf{G}_0$ to game $\mathbf{G}_1$ so that $\mathcal{RO}_n$ returns a random value when $\mathcal{A}_1$ poses a message that is posed to $\mathcal{RO}_n$ by Enc to generate the challenge ciphertext. In game $\mathbf{G}_1$, outputs of $\mathcal{RO}_n$ does not contain any information about computations to generate the challenge ciphertext for $\mathcal{A}_1$. Thirdly, we transform $\mathbf{G}_1$ to game $\mathbf{G}_2$ so that ciphertext $\mathbf{c}$ is generated from all zero messages instead of given messages from $\mathcal{A}_1$. In game $\mathbf{G}_2$, ciphertext $\mathbf{c}$ does not contain any information about outputs of $\mathcal{A}_1$. Thus, $\mathcal{A}_1$ cannot hand over any information to $\mathcal{A}_2$ with $\mathbf{c}$. Fourthly, we transform $\mathbf{G}_2$ to game $\mathbf{G}_3$ so that the table of inputs and outputs of each oracle in $\mathcal{WRO}$ (except $\mathcal{RO}_n$) for $\mathcal{A}_1$ is independent of the table for $\mathcal{A}_2$ according to the output of $\mathcal{A}_1$. In game $\mathbf{G}_3$, queries to sub-oracles for $\mathcal{A}_2$ does not contain any information about the output of $\mathcal{A}_1$, and $\mathcal{A}_1$ cannot hand over any information to $\mathcal{A}_2$ with sub-oracles. Finally, we transform $\mathbf{G}_3$ to game $\mathbf{G}_4$ so that $\mathcal{RO}_n$ returns a random value when $\mathcal{A}_2$ poses a message that is posed to $\mathcal{RO}_n$ by Enc to generate the challenge ciphertext. In game $\mathbf{G}_4$, outputs of $\mathcal{RO}_n$ does not contain any information about computations to generate the challenge ciphertext for $\mathcal{A}_2$. Thus, the advantage of $\mathcal{A}_2$ in $\mathbf{G}_4$ is nothing.

Proof. We denote $\mathsf{Adv}(\mathcal{A}, \mathbf{G}_i)$ by the advantage of adversary $(\mathcal{A}_1, \mathcal{A}_2)$ when participating in experiment $\mathbf{G}_i$. We start with game $\mathbf{G}_0$ which is exactly the same game as the ID-CDA game in the $\mathcal{WRO}$ model. It means $\mathsf{Adv}(\mathcal{A}, \mathbf{G}_0) = \mathsf{Adv}^{\text{id-cda}}_{\mathsf{IDREwH1},\mathcal{WRO}}(\mathcal{A}_1, \mathcal{A}_2)$.

```
$\underline{\mathsf{SimC}'_{\text{main}}}$
$\beta'' \xleftarrow{\$} \{0,1\}$
receive $id^*$ from $\mathcal{A}_2$
send $id^*$ to $\mathcal{CH}$
receive $params$ from $\mathcal{CH}$
$(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r}) \leftarrow \mathcal{A}_1$
send $(\mathbf{m}''_\beta, \mathbf{0})$ to $\mathcal{CH}$
receive $\mathbf{c}$ from $\mathcal{CH}$
send $(params, \mathbf{c})$ to $\mathcal{A}_2$
receive $\beta'$ from $\mathcal{A}_2$
return 0 if $\beta' = \beta''$
       1 otherwise
```

```
$\underline{\mathsf{SimC}'_{\mathsf{IBE.Gen}}(id)}$
send $id$ to IBE.Gen oracle
receive $sk_{id}$ from IBE.Gen oracle
return $sk_{id}$

$\underline{\mathsf{SimC}'_{\mathcal{RO}_n}(M)}$
send $M$ to $\mathcal{RO}$
receive $\mathsf{F}[M]$ from $\mathcal{RO}$
return $\mathsf{F}[M]$

$\underline{\mathsf{SimC}'_{\mathcal{RO}^*_v}(M)}$
If $\mathsf{F}^*[M] = \perp$, $\mathsf{F}^*[M] \xleftarrow{\$} \{0,1\}^v$
return $\mathsf{F}^*[M]$;

$\underline{\mathsf{SimC}'_{\mathcal{RO}^\dagger_w}(M)}$
If $\mathsf{F}^\dagger[M] = \perp$ then $\mathsf{F}^\dagger[M] \xleftarrow{\$} \{0,1\}^w$
return $\mathsf{F}^\dagger[M]$;
```

```
$\underline{\mathsf{SimC}'_{\mathcal{TO}}(y)}$
If $\exists_1 M$ s.t. $\mathsf{F}^\dagger[M] = y$ then return $M$
Otherwise return $\perp$

$\underline{\mathsf{SimC}'_{E}(k, x)}$
If $\mathsf{E}[k, x] = \perp$,
    $y \xleftarrow{\$} \{0,1\}^b \backslash T^+[k]$
    $\mathsf{E}[k, x] \leftarrow y, \mathsf{D}[k, y] \leftarrow x$,
    $T^+[k] \xleftarrow{\cup} \{y\}$, $T^-[k] \xleftarrow{\cup} \{x\}$
return $\mathsf{E}[k, x]$

$\underline{\mathsf{SimC}'_{D}(k, y)}$
If $\mathsf{D}[k, y] = \perp$,
    $x \xleftarrow{\$} \{0,1\}^b \backslash T^-[k]$;
    $\mathsf{E}[k, x] \leftarrow y, \mathsf{D}[k, y] \leftarrow x$,
    $T^+[k] \xleftarrow{\cup} \{y\}$, $T^-[k] \xleftarrow{\cup} \{x\}$
return $\mathsf{D}[k, y]$
```

Fig. 16. Simulation SimC' by adversary $\mathcal{C}$

Game $\mathbf{G}_1$: $\mathcal{RO}_n$ returns a random value if the following event occurs:

- Bad_1 : $\mathcal{A}_1$ poses $(params, id^*, m_\beta, r)$ to $\mathcal{RO}_n$ where $(params, id^*, m_\beta, r)$ is posed to $\mathcal{RO}_n$ by Enc to generate the challenge ciphertext.

All other procedures are computed as the same way in $\mathbf{G}_0$.

Lemma 8. $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_1) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_0)| \leq q_{\mathcal{RO}} \cdot \mathsf{maxparams}_{\mathcal{IBE}_r}$.

Proof. The difference between $\mathbf{G}_0$ and $\mathbf{G}_1$ only occurs in Bad_1. From Difference Lemma [22], we have that $|\mathsf{Adv}(\mathcal{B}, \mathbf{G}_1) - \mathsf{Adv}(\mathcal{B}, \mathbf{G}_0)| \leq \Pr[\mathsf{Bad}_1]$.

We estimate $\Pr[\mathsf{Bad}_1]$. Since $params$ is not given for $\mathcal{A}_1$ and is included in each query to $\mathcal{RO}_n$ by Enc, the only way to pose $(params, id^*, m_\beta, r)$ to $\mathcal{RO}_n$ is choosing $params$ randomly $q_{\mathcal{RO}}$ times. We have that $\Pr[\mathsf{Bad}_1] \leq q_{\mathcal{RO}} \cdot \mathsf{maxparams}_{\mathcal{IBE}_r}$. □

Game $\mathbf{G}_2$: Challenge ciphertext $\mathbf{c} \leftarrow \mathsf{IBE.Enc}_r(params, id^*, \mathbf{m}_\beta; \mathcal{RO}_n(params, id^*, \mathbf{m}_\beta, \mathbf{r}))$ is replaced with $\mathbf{c} \leftarrow \mathsf{IBE.Enc}_r(params,\ id^*, \mathbf{0}; \mathcal{RO}_n(params, id^*, \mathbf{0}, \mathbf{r}))$ where $\mathbf{0}$ is a vector of l zero strings of length λ. All other procedures are computed as the same way in $\mathbf{G}_1$.

Lemma 9. $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_2) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_1)| \leq 2\mathsf{Adv}^{\text{id-cpa}}_{\mathcal{IBE}_r, \mathcal{RO}}(\mathcal{C})$.

Proof. We show that if $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_2) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_1)|$ is non-negligible, we can construct an adversary $\mathcal{C}$ breaking ID-CPA security of $\mathcal{IBE}_r$ in the RO model. Fig. 16 shows simulation $\mathsf{SimC}' = (\mathsf{SimC}'_{\text{main}}, \mathsf{SimC}'_{\mathsf{IBE.Gen}}, \mathsf{SimC}'_{\mathcal{RO}}, \mathsf{SimC}'_{\mathcal{RO}^*}, \mathsf{SimC}'_{\mathcal{RO}^\dagger}, \mathsf{SimC}'_{\mathcal{TO}}, \mathsf{SimC}'_{E}, \mathsf{SimC}'_{D})$ by $\mathcal{C}$ respectively.

$\mathcal{C}$ simulates all queries to $\mathcal{WRO}$ for $\mathcal{A}_1$ and $\mathcal{A}_2$ with simulation SimC'. SimC' is identical with the definition of $\mathcal{WRO}$. Thus, $\mathcal{A}$ cannot distinguish game $\mathbf{G}_1$ and $\mathbf{G}_2$ from the simulation on the interface of $\mathcal{WRO}$. If $\beta = 1$ in ID-CPA game for $\mathcal{IBE}_r$, it is clear that all interfaces for $\mathcal{A}$ is exactly same as game $\mathbf{G}_2$. If $\beta = 0$ in ID-CPA game for $\mathcal{IBE}_r$, it is clear that all interfaces for $\mathcal{A}$ is exactly same as game $\mathbf{G}_1$ if $\beta = \beta''$.

Therefore, if $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_1) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_0)|$ is non-negligible, $\mathcal{C}$ also breaks ID-CPA security of $\mathcal{IBE}_r$ if $\beta = \beta''$ (i.e., with probability 1/2). We have that $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_2) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_1)| \leq 2\mathsf{Adv}^{\text{id-cpa}}_{\mathcal{IBE}_r,\mathcal{RO}}(\mathcal{C})$. □

Game $\mathbf{G}_3$: When $\mathcal{A}_2$ poses a query related to $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$ (which is the output of $\mathcal{A}_1$) to $\mathcal{RO}^*_v, \mathcal{RO}^\dagger_w, \mathcal{TO}$ or $\mathsf{IC}_{a,b} = (E, D)$, then outputs are randomly chosen. That is, tables $\mathsf{F}^*, \mathsf{F}^\dagger, \mathsf{E}$ and D are not preserved for $\mathcal{A}_1$ and $\mathcal{A}_2$ according to $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$. All other procedures are computed as the same way in $\mathbf{G}_2$.

Lemma 10. $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_3) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_2)| \leq \frac{4q^2_{\mathcal{RO}^*} + 4q^2_{\mathcal{RO}^\dagger}}{2^\mu} + max\left\{\frac{4q^2_{\mathcal{TO}}}{2^\mu}, \frac{4q^2_{\mathcal{TO}}}{2^w}\right\} + max\left\{\frac{4q^2_E + 4q^2_D}{2^\mu}, \frac{4q^2_E + 4q^2_D}{2^b}\right\}$.

Proof. The difference between $\mathbf{G}_2$ and $\mathbf{G}_3$ only occurs when $\mathcal{A}_1$ and $\mathcal{A}_2$ poses a same query related to $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$ to $\mathcal{RO}^*_v, \mathcal{RO}^\dagger_w, \mathcal{TO}$ or $\mathsf{IC}_{a,b} = (E, D)$. We denote the event that a common query related to $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$ is posed to $\mathcal{RO}^*_v$ by $\mathcal{A}_1$ and $\mathcal{A}_2$ as $\mathsf{Bad}_{\mathcal{RO}^*}$. Similarly, we define events $\mathsf{Bad}_{\mathcal{RO}^\dagger}$, $\mathsf{Bad}_{\mathcal{TO}}$, Bad_E, and Bad_D. From Difference Lemma [22], we have that $|\mathsf{Adv}(\mathcal{B}, \mathbf{G}_3) - \mathsf{Adv}(\mathcal{B}, \mathbf{G}_2)| \leq \Pr[\mathsf{Bad}_{\mathcal{RO}^*} \vee \mathsf{Bad}_{\mathcal{RO}^\dagger} \vee \mathsf{Bad}_{\mathcal{TO}} \vee \mathsf{Bad}_E \vee \mathsf{Bad}_D] \leq \Pr[\mathsf{Bad}_{\mathcal{RO}^*}] + \Pr[\mathsf{Bad}_{\mathcal{RO}^\dagger}] + \Pr[\mathsf{Bad}_{\mathcal{TO}}] + \Pr[\mathsf{Bad}_E] + \Pr[\mathsf{Bad}_D]$.

In game $\mathbf{G}_2$ and $\mathbf{G}_3$, ciphertext $\mathbf{c}$ does not give any information about $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$ and queries to $\mathcal{WRO}$ by $\mathcal{A}_1$ to $\mathcal{A}_2$. On queries to $\mathcal{RO}_n$, interfaces of $\mathcal{A}_2$ in $\mathbf{G}_2$ and $\mathbf{G}_3$ are identical. Thus, the only way to pose such a query is guessing under min-entropy μ, or the output length w of $\mathcal{RO}_T$ and the output length b of (E, D). According to the birthday paradox, for oracles $\mathcal{RO}^*$ and $\mathcal{RO}^\dagger$ the probability of collisions in guessing is at most $(2q_{\mathcal{RO}^*})^2/2^\mu$, and $(2q_{\mathcal{RO}^\dagger})^2/2^\mu$, respectively. Also, for oracle $\mathcal{TO}$ the probability of collision in guessing is at most $(2q_{\mathcal{TO}})^2/2^\mu$ if $\mu < w$, $(2q_{\mathcal{TO}})^2/2^w$ otherwise, and for oracles E and D the probability of collisions in guessing is at most $(2q_E)^2/2^\mu$ and $(2q_D)^2/2^\mu$ if $\mu < b$, $(2q_E)^2/2^b$ and $(2q_D)^2/2^b$ otherwise. Therefore, $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_3) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_2)| \leq (4q^2_{\mathcal{RO}^*} + 4q^2_{\mathcal{RO}^\dagger})/2^\mu + max\left\{4q^2_{\mathcal{TO}}/2^\mu, 4q^2_{\mathcal{TO}}/2^w\right\} + max\left\{(4q^2_E + 4q^2_D)/2^\mu, (4q^2_E + 4q^2_D)/2^b\right\}$. □

Game $\mathbf{G}_4$: $\mathcal{RO}_n$ returns a random value if the following event occurs:

- Bad_2 : $\mathcal{A}_2$ poses $(params, id^*, m_\beta, r)$ to $\mathcal{RO}_n$ where $(params, id^*, m_\beta, r)$ is posed to $\mathcal{RO}_n$ by Enc to generate the challenge ciphertext.

All other procedures are computed as the same way in $\mathbf{G}_3$.

Lemma 11. $|\mathsf{Adv}(\mathcal{A}, \mathbf{G}_4) - \mathsf{Adv}(\mathcal{A}, \mathbf{G}_3)| \leq \frac{q_{\mathcal{RO}}}{2^\mu}$.

Proof. The difference between $\mathbf{G}_3$ and $\mathbf{G}_4$ only occurs in Bad_2. From Difference Lemma [22], we have that $|\mathsf{Adv}(\mathcal{B}, \mathbf{G}_4) - \mathsf{Adv}(\mathcal{B}, \mathbf{G}_3)| \leq \Pr[\mathsf{Bad}_2]$.

We estimate $\Pr[\mathsf{Bad}_2]$. Because $\mathcal{RO}_n$ is a truly random function, ciphertext $\mathbf{c}$ is replaced, and $\mathcal{A}_1$ does not pose related queries to $(\mathbf{m}_\beta, \mathbf{r})$ to sub-oracles, $\mathcal{A}_2$ cannot obtain more information of $(\mathbf{m}_\beta, \mathbf{r})$ than min-entropy μ. Thus, the only way to pose $(params, id^*, m_\beta, r)$ to $\mathcal{RO}_n$ is guessing (m_β, r) under min-entropy μ $q_{\mathcal{RO}}$ times. We have that $\Pr[\mathsf{Bad}_2] \leq \frac{q_{\mathcal{RO}}}{2^\mu}$. □

We estimate $\mathsf{Adv}(\mathcal{A}, \mathbf{G}_4)$. Ciphertext $\mathbf{c}$ does not give any information about $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$. Also, outputs of $\mathcal{WRO}$ is independent of $(\mathbf{m}_0, \mathbf{m}_1, \mathbf{r})$ for $\mathcal{A}_2$. Thus, the only way to win in game $\mathbf{G}_4$ is randomly guessing β. Therefore, $\mathsf{Adv}(\mathcal{A}, \mathbf{G}_4) = 0$.

To conclude, we have $\mathsf{Adv}^{\text{id-cda}}_{\mathsf{IDREwH1},\mathcal{WRO}}(\mathcal{A}_1, \mathcal{A}_2) \leq 2\mathsf{Adv}^{\text{id-cpa}}_{\mathcal{IBE}_r,\mathcal{RO}}(\mathcal{C}) + q_{\mathcal{RO}} \cdot \mathsf{maxparams}_{\mathcal{IBE}_r} + (q_{RO} + 4q^2_{\mathcal{RO}^*} + 4q^2_{\mathcal{RO}^\dagger})/2^\mu + max\left\{4q^2_{\mathcal{TO}}/2^\mu, 4q^2_{\mathcal{TO}}/2^w\right\} + max\left\{(4q^2_E + 4q^2_D)/2^\mu, (4q^2_E + 4q^2_D)/2^b\right\}$. □