Security of Practical Cryptosystems Using Merkle-Damgård Hash Function in the Ideal Cipher Model

Yusuke Naito[1], Kazuki Yoneyama[2], Lei Wang[3], and Kazuo Ohta[3]

[1] Mitsubishi Electoric Corporation
[2] NTT Corporation
[3] The University of Electro-Communications

Abstract. Since the Merkle-Damgård (MD) type hash functions are differentiable from ROs even when compression functions are modeled by ideal primitives, there is no guarantee as to the security of cryptosystems when ROs are instantiated with structural hash functions. In this paper, we study the security of the instantiated cryptosystems whereas the hash functions have the well known structure of Merkle-Damgård construction with Stam's type-II compression function (denoted MD-TypeII) in the Ideal Cipher Model (ICM). Note that since the Type-II scheme includes the Davies-Meyer compression function, SHA-256 and SHA-1 have the MD-TypeII structure.

We show that OAEP, RSA-KEM, PSEC-KEM, ECIES-KEM and many other encryption schemes are secure when using the MD-TypeII hash function. In order to show this, we customize the indifferentiability framework of Maurer, Renner and Holenstein. We call the customized framework "indifferentiability with condition". In this framework, for some condition α that cryptosystem C satisfies, if hash function H is indifferentiable from RO under condition α, C is secure when RO is instantiated with H. We note the condition of "prefix-free" that the above schemes satisfy. We show that the MD-TypeII hash function is indifferentiable from RO under this condition. When the output length of RO is incompatible with that of the hash function, the output size is expanded by Key Derivation Functions (KDFs). Since a KDF is specified as MGF1 in RSA's PKCS #1 V2.1, its security discussion is important in practice. We show that, KDFs using the MD-TypeII hash function (KDF-MD-TypeII) are indifferentiable from ROs under this condition of "prefix-free". Therefore, we can conclude that the above practical encryption schemes are secure even when ROs are instantiated with (KDF-)MD-TypeII hash functions.

Dodis, Ristenpart and Shrimpton showed that FDH, PSS, Fiat-Shamir, and so on are secure when RO is instantiated with the MD-TypeII hash function in the ICM, their analyses use the different approach from our approach called indifferentiability from public-use RO (pub-RO). They showed that the above cryptosystems are secure in the pub-RO model and the MD-TypeII hash function is indifferentiable from pub-RO. Since their analyses did not consider the structure of KDFs, there might exist some attack using a KDF's structure. We show that KDFs using pub-RO (KDF-pub-RO) is differentiable from pub-RO. Thus, we cannot trivially extend the result of Dodis et al to the indifferentiability for KDF-MD-TypeII hash functions. We propose a new oracle called private interface leak RO (privleak-RO). We show that KDF-pub-ROs are indifferentiable from privleak-ROs and the above cryptosystems are secure in the privleak-RO model. Therefore, by combining the result of Dodis et al. with our result, we can conclude that the above cryptosystems are secure when ROs are instantiated with KDF-MD-TypeII hash functions.

Since OAEP, RSA-KEM, PSEC-KEM, ECIES-KEM and many other encryption schemes are insecure in the pub-RO (privleak-RO) model, we cannot confirm the security of these encryption schemes from the approach of Dodis et al. Therefore, the result of Dodis et al can be supplemented with our result. Consequently, from the two results we can confirm the security of almost practical cryptosystems when ROs are instantiated with (KDF-)MD-TypeII hash functions.

Keywords: Indifferentiability with condition, weakened random oracle, Merkle-Damgård, type-II compression function, Davies-Meyer, PGV, key-derivation functions, OAEP, RSA-KEM, PSEC-KEM, ECIES-KEM.

1 Introduction

The Random Oracle (RO) Methodology is a well known methodology for designing efficient cryptosystems and many important cryptosystems have been designed on RO methodology. For example, RSA-OAEP [3], RSA-PSS [3], RSA-KEM [37], PSEC-KEM [37], and ECIES-KEM [37], which are standardized in RSA's PKCS #1 V2.1 or ISO 18033-2, are designed by this methodology. In this methodology, hash functions are viewed as ROs. When implementing a cryptosystem, RO is instantiated by a cryptographic hash function such as SHA-2 family and SHA-1 [31]. However, since there are several separation results for ROs and cryptographic hash functions [11], the heuristic evidence of the methodology is questionable.

In order to fill the theoretical gap, Coron, Dodis, Malinaud, and Puniya [15] introduced a new property of hash functions called indifferentiability from RO. In this property, while underlying primitive P (e.g. compression function) is in the ideal model, if hash function H^P, which is constructed from P, is indifferentiable from RO, we can use H^P as an RO. Namely, this property fills the structural gap between hash functions and ROs while underlying primitives follow ideal models.

The popular hash functions are SHA-2 family hash functions (e.g. SHA-256 and SHA-512) that are published as FIPS standard. These hash functions use the Merkle-Damgård (MD) structure [17, 29] and the Davies-Meyer compression function (DMCF). While the MD hash function with DMCF (DMMDHF) offers collision resistance in the Ideal Cipher Model (ICM) [5], the DMMDHF is differentiable from RO due to the extension attack. The attack is that for DMMDHF H, $H(m_1||m_2)$ is calculated from $H(m_1)$ and m_2. On the other hand, the attack cannot be applied to ROs. Due to the state of differentiability, there is no guarantee as to the security of cryptosystems when RO is instantiated with DMMDHF. This leaves open the question whether or not cryptosystems can be securely instantiated when RO is replaced by DMMDHF.

Dodis, Restinpart and Shrimpton answered the question for several cryptosystems [19]. They proved that several cryptosystems are secure when RO is instantiated with a MD hash function that use Stam's Type-II compression function [39] (denoted MD-TypeII) in the ICM. Note that since the Type-II scheme includes DMCF (and also several PGV schemes [35, 5]), the MD-TypeII hash function includes DMMDHF. In order to prove the security, they proposed the Weakened Random Oracle (WRO) approach. This approach states that for hash function H (1) define a WRO such that H is indifferentiable from WRO and (2) prove the security of cryptosystems in the WRO model. They defined public-use Random Oracle (pub-RO) that leaks the hash list of a random oracle. They showed that the MD-typeII hash function is indifferentiable from pub-RO. Since adversaries know all inputs of random oracles for FDH [2], PFDH [14], Fiat-Shamir [20], BLS [8], PSS [4], a variant of Boneh-Franklin IBE [36] and Boneh-Boyern IBE [9], the additional function of pub-RO does not leak any useful information to the adversaries. Therefore, these cryptosystems are secure in the pub-RO model. Thus these cryptosystems are secure when RO is instantiated with the MD-typeII hash function. We call these cryptosystems "pub-RO secure cryptosystems".

Open Problems. While many cryptosystems are secure when RO is instantiated with the MD-TypeII hash function, the security of the following important cryptosystems remains unclear.

1. Since OAEP [4], RSA-KEM [37], PSEC-KEM [37], ECIES-KEM [37] and many other encryption schemes are insecure in the pub-RO model [40, 30], the result of Dodis et al. [19]

provide no support for the security of these cryptosystems with the MD-TypeII hash function. Therefore, the security of these important encryption schemes remains an open problem.

2. When RO has longer output length than the hash function, RO is instantiated by the Key Derivation Function (KDF) [37]. Note that KDFs include MGF1 [26], Bellare-Rogaway 96 scheme [4] and so on. While Dodis et al. proved that pub-RO secure cryptosystems are secure when RO is instantiated with the MD-TypeII hash function, they did not consider the KDF's structure. Therefore, the security of these cryptosystems using KDFs remains an open problem, since there might exist some attack based on the KDF's structure.

Security of Encryption Schemes. First, we show that OAEP, RSA-KEM, PSEC-KEM, ECIES-KEM, and many other encryption schemes (e.g. OAEP+ [38], SAEP [7], SAEP+ [7], and many other schemes [1, 13, 16, 18, 25, 24, 33, 34]) are secure in ICM when using the MD-typeII hash function and KDFs with MD-typeII hash functions (denote KDF-MD-typeII). To confirm the security of these encryptions, we customize the indifferentiability framework of Maurer, Renner and Holenstein [28]. We call the customized framework *indifferentiability with condition.* In this framework, we consider some condition α that cryptosystem C satisfies. If hash function H is indifferentiable from RO under condition α, C is secure when RO is replaced by H. α is the condition of inputs to H. Namely, we say that "cryptosystem C satisfies condition α" if all input values from C to H satisfy condition α and "H is indifferentiable from RO under condition α" if H is indifferentiable from RO when all queries from any distinguisher to H/RO satisfy condition α. We introduce the following procedure to confirm the security of the cryptosystems.

1. Identify condition α that the cryptosystems satisfy.
2. Prove that the (KDF-)MD-typeII hash function is indifferentiable from RO under condition α.

Step 1: We note the condition of the encryption schemes: the input size of the hash function is fixed. Namely, all input values, x, x', of the hash function satisfy $|x| = |x'|$. For any different two values x, x' that yield $|x| = |x'|$, x is not a prefix of x', the encryption schemes satisfy the condition "prefix-free". Therefore, we use the condition "prefix-free".

Step 2: In order to prove that the (KDF-)MD-typeII hash functions are indifferentiable from ROs under the condition "prefix-free", we propose the following approach. Let H be a hash function and pfpad be any prefix-free padding function.

- If $H \circ \mathsf{pfpad}$ is indifferentiable from RO, H is indifferentiable from RO under the condition "prefix-free" where $H \circ \mathsf{pfpad}$ is a hash function with prefix-free padding.
- $H \circ \mathsf{pfpad}$ is indifferentiable from RO.

The first item implies that the result of the indifferentiability for $H \circ \mathsf{pfpad}$ can be transformed into the result of the indifferentiability with condition for H. From the second item, we can conclude that H is indifferentiable from RO under the condition "prefix-free". We show that the (KDF-)MD-TypeII hash functions with any prefix-free padding are indifferentiable from ROs. Therefore, the (KDF-)MD-TypeII hash functions are indifferentiable from ROs under the condition "prefix-free".

The above two steps allow us to conclude that OAEP, RSA-KEM, PSEC-KEM, ECIES-KEM and many other encryption schemes are secure when ROs are instantiated with the (KDF-)MD-TypeII hash function. Several papers [10, 6, 23, 32] showed that padding-based encryption schemes (e.g., OAEP) are provably unprovable in the standard model when using a black-box reduction. Namely, the encryption schemes are provably unprovable when considering "full" structures of hash functions. Our result shows that the security of the encryption schemes are provable when considering structures of the (KDF-)MD-TypeII hash functions except for block ciphers. That is, our result shows that there is no generic attack on the encryption schemes that use (KDF-)MD-TypeII hash functions that treat block ciphers like ideal ciphers.

Security of Pub-RO Secure Cryptosystems Using KDF-MD-TypeII Hash Functions. By using the WRO approach, we show that the pub-RO secure cryptosystems are secure when ROs are instantiated with KDF-MD-TypeII hash functions in the ICM. First we show that KDFs using pub-RO are differentiable from pub-RO. Thus we cannot simply extend the result of Dodis et al. to the indifferentiability for the KDF-MD-TypeII hash functions. Therefore we propose a new WRO called private interface leaking RO (privleak-RO). The oracle leaks all input-output pairs of a private interface of RO that are used in cryptosystem calculations but does not leak input-output pairs of the public interface. Since adversaries know all inputs of the random oracles in pub-RO secure cryptosystems, these cryptosystems are secure even when ROs replaced by privleak-ROs. We show that KDFs using pub-ROs are indifferentiable from privleak-ROs. Since MD-typeII hash functions are indifferentiable from pub-RO, the KDF-MD-typeII hash functions are indifferentiable from privleak-ROs. As a result, pub-RO secure cryptosystems are secure when RO is instantiated with the KDF-MD-typeII hash function.

Related Works. Leurent and Nguyen [27] studied the security of cryptosystems when ROs are replaced with KDFs that use weakened hash functions such as SHA-1 and MD5. They showed that these hash functions offer much lower security than the theoretical security of RO. For example, when the output length of RO is 1024 bits, a collision of KDF3 using MD5 is found with 2^{106} MD5 computations and a preimage is found with 2^{166} MD5 computations. They also examined the security of padding-based signature schemes when ROs are replaced with the weakened hash functions. They showed that for several signature schemes a collision of a hash function can be transformed into a key recovery attack. Their analyses examined the case of weakened hash functions. Our analyses examine the case of secure hash functions.

Coron, Dodis, Malinaud and Puniya [15], Chang, Lee, Nandi and Yung [12], and Gong, Lai and Chen [21] proved that the MD hash functions with any prefix-free padding with several PGV schemes are indifferentiable from ROs. However, these results don't imply that cryptosystems satisfying the "prefix-free" condition are secure when ROs are instantiated with MD hash functions *without* prefix-free padding. The result of the first point of the above step 2 is needed to prove the security of the cryptosystems. Note that by using the above first point, these indifferentiability results can be transformed into a proof of indifferentiability with condition.

Naito, Yoneyama, Wang and Ohta [30] defined Extension Attack Simulatable Random Oracle (ERO) to which the extension attack can be applied. They showed that the MD hash function in the fixed input length (FIL) RO model is indifferentiable from ERO and OAEP, its variants and RSA-KEM are secure in the ERO model. Since the Type-II scheme is differentiable from FILRO, the result cannot be transformed into a proof of indifferentiable for the (KDF-)MD-TypeII hash functions.

2 Preliminaries

Notation. For two values x, y, $x||y$ is the concatenated value of x and y. $x \leftarrow y$ means assigning y to x. $\oplus$ is bitwise exclusive or. $|x|$ is the bit length of value x. $\langle i \rangle$ is the 64 bit value encoded as a string of i. For set (list) $\mathcal{T}$ and element W, $\mathcal{T} \leftarrow W$ means to insert W into $\mathcal{T}$ (if W is already inserted in $\mathcal{T}$, W is not inserted.). For some jn bit value x, let $x[1], \ldots, x[j]$ be n bit values of each block of x (namely $x = x[1]|| \cdots ||x[j]$). For some value x, $x_{[w]}$ is the last w bit value of x and $x_{(w)}$ is the first $|x| - w$ bit value of x (namely $x = x_{(w)}||x_{[w]}$). $\mathcal{C}_{d,n} = (E, D)$ be a ideal cipher where $E : \{0,1\}^d \times \{0,1\}^n \rightarrow \{0,1\}^n$ is an encryption oracle, $D : \{0,1\}^d \times \{0,1\}^n \rightarrow \{0,1\}^n$ is a decryption oracle, the key size is d bit and the cipher text size is n bit. $\mathcal{F}_b : \{0,1\}^* \rightarrow \{0,1\}^b$ is a random oracle.

Indifferentiability Framework [28]. Let $\mathcal{U}$ and $\mathcal{W}$ be some primitives. In this framework, we consider two interfaces: public interface and private interface. Honest parties (e.g. cryptosystems) can access the private interface and adversaries can access the public interface. The private interface of $\mathcal{U}$ ($\mathcal{W}$) denotes $\mathcal{U}^{priv}$ ($\mathcal{W}^{priv}$) and the public interface of $\mathcal{U}$ ($\mathcal{W}$) denotes $\mathcal{U}^{pub}$ ($\mathcal{W}^{pub}$). We consider two experiments. Let A be any distinguisher. One is that A accesses to $\mathcal{W}^{priv}$ and $\mathcal{W}^{pub}$. Another is that A accesses to $\mathcal{U}^{priv}$ and a simulator S that simulates $\mathcal{W}^{pub}$ by accessing $\mathcal{U}^{pub}$. The definition of indifferentiability is as follows.

Definition 1. *$\mathcal{W}$ is (t_A, t_S, ϵ)-indifferentiable from $\mathcal{U}$, if there exists S of running time at most t_S for any A of running time at most t_A such that*

$$|Pr[A^{\mathcal{W}^{priv}, \mathcal{W}^{pub}} \Rightarrow 1] - Pr[A^{\mathcal{U}^{priv}, S(\mathcal{U}^{pub})} \Rightarrow 1]| \leq \epsilon. \tag{1}$$

We denote "$\mathcal{W}$ is indifferentiable from $\mathcal{U}$" by $\mathcal{W} \sqsubset \mathcal{U}$.

We say "$\mathcal{W}$ is indifferentiable from $\mathcal{U}$" or $\mathcal{W} \sqsubset \mathcal{U}$ when ϵ is negligible. From the definition, the following lemma is obtained.

Lemma 1. *If $\mathcal{W} \sqsubset \mathcal{U}$, then for any cryptosystem $\mathcal{C}$ $\mathcal{C}(\mathcal{W})$ is at least as secure as $\mathcal{C}(\mathcal{U})$. We denote "$\mathcal{C}(\mathcal{W})$ is at least as secure as $\mathcal{C}(\mathcal{U})$" by $\mathcal{C}(\mathcal{W}) \succ \mathcal{C}(\mathcal{U})$.*

$\mathcal{C}(\mathcal{W}) \succ \mathcal{C}(\mathcal{U})$ means that if $\mathcal{C}(\mathcal{W})$ is secure then $\mathcal{C}(\mathcal{U})$ is also secure. The definition of $\mathcal{C}(\mathcal{W}) \succ \mathcal{C}(\mathcal{U})$ is as follows.

Definition 2. *$\mathcal{C}(\mathcal{W}) \succ \mathcal{C}(\mathcal{U})$ if for all environments Env (distinguisher of $\mathcal{C}$) the following holds: For any attacker $\mathcal{A}$ accessing $\mathcal{C}(\mathcal{W}^{priv})$ and $\mathcal{W}^{pub}$ there exists an attacker $\mathcal{A}'$ accessing $\mathcal{C}(\mathcal{U}^{priv})$ and $\mathcal{U}^{pub}$ such that $|Pr[Env^{\mathcal{C}(\mathcal{W}^{priv}), \mathcal{A}} \Rightarrow 1] - Pr[Env^{\mathcal{C}(\mathcal{U}^{priv}), \mathcal{A}'} \Rightarrow 1]$ is negligible in the security parameter of $\mathcal{C}$.*

In the experiment of indifferentiability of hash function H^P using a primitive P from RO, A interacts with (H^P, P) or (RO, S^{RO}). Simulator S simulates P.

Merkle-Damgård. Let $h : \{0,1\}^{d+n} \rightarrow \{0,1\}^n$ be a compression function using primitive P (more strictly h^P) and $\mathsf{pad} : \{0,1\}^* \rightarrow (\{0,1\}^d)^*$ be a padding function. We define Merkle-Damgård hash function MD^h as follows where IV is an n-bit initial value.

```
$\underline{\mathrm{MD}^h(M)}$
$z[0] \leftarrow IV$;
Break $\mathsf{pad}(M)$ into $d$-bit blocks, $\mathsf{pad}(N) = M[1]|| \cdots ||M[l]$;
for $i = 1, \ldots, l$ do $z[i] \leftarrow h(z[i-1], M[i])$;
Ret $z[l]$;
```

We write MD^h, when padding pad is the prefix-free padding pfpad, by PFMD^h.

Generalized Rate-1 Block-cipher-based Compression Function [17, 29, 35]. Stam generalized rate-1 block-cipher-based compression functions [39, 35]. He considered compression functions $\mathsf{SCF}^{\mathcal{C}_{d,n}}$ that, on input of chaining variable $v \in \{0,1\}^n$ and message block $m \in \{0,1\}^d$, operates as follows where $C^{\mathrm{PRE}} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^d \times \{0,1\}^n$ and $C^{\mathrm{POST}} : \{0,1\}^d \times \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ are functions called preprocessing and postprocessing, respectively.

```
$\underline{\mathsf{SCF}^{\mathcal{C}_{d,n}}(v,m)}$
$(k,x) \leftarrow C^{\mathrm{PRE}}(v,m)$;
$y \leftarrow E(k,x)$
Ret $w \leftarrow C^{\mathrm{POST}}(v,m,y)$;
```

He also defined auxiliary post-processing function $C^{\mathrm{AUX}} : \{0,1\}^d \times \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ such that $C^{\mathrm{AUX}}(k,x,y) = C^{\mathrm{POST}}(v,m,y)$. Stam defined a Type-II scheme iff Stam defined Type-II block-cipher-based compression function [39]. Compression function SCF is the Type-II scheme if: 1) C^{PRE} is bijective, 2) for all v, m $C^{\mathrm{POST}}(v,m,\cdot)$ is bijective, and 3) for all k, the inverse map $C_1^{-\mathrm{PRE}}(k,\cdot)$ is bijective. Here the map $C_1^{-\mathrm{PRE}} : \{0,1\}^d \times \{0,1\}^n \to \{0,1\}^n$ is defined by $C_1^{-\mathrm{PRE}}(k,m) = v$ where $(v,m) = C^{-\mathrm{PRE}}(k,x)$. The Type-II scheme includes the Group-2 PGV schemes and 8 Group PGV schemes (e.g. Davies-Meyer) in [5, 35]. The Davies-Meyer has $C^{\mathrm{PRE}}(v,m) = (m,v)$, $C^{\mathrm{POST}}(v,m,y) = v \oplus y$ and $C^{\mathrm{AUX}}(k,x,y) = x \oplus y$.

KDFs [37]. Let $H : \{0,1\}^* \to \{0,1\}^n$ be a hash function. KDF1, KDF2 and KDF3 are defined by $\mathsf{KDF1}\text{-}H(M) = H(M||\langle 0\rangle)||H(M||\langle 1\rangle)||\ldots$, $\mathsf{KDF2}\text{-}H(M) = H(M||\langle 1\rangle)||H(M||\langle 2\rangle)||\ldots$, and $\mathsf{KDF3}\text{-}H(M) = H(\langle 0\rangle||M)||H(\langle 1\rangle||M)||\ldots$.

Public-use Random Oracle [19]. Pub-RO consists of RO $\mathcal{F}_b$ and Leak Oracle (LO) $\mathcal{F}_{leak}$ that leaks the RO list. The description is as follows where $\mathcal{F}_b$ is a RO whose the output size is b bit and $\mathcal{F}_{leak}$ is a LO.

```
$\underline{\mathcal{F}_b(M)}$
If $\mathsf{F}_b(M) \neq \perp$, ret $\mathsf{F}_b(M)$;
$\mathsf{F}_b(M) \xleftarrow{\$} \{0,1\}^n$;
$L_{leak} \leftarrow (M, \mathsf{F}_b(M))$;
Ret $\mathsf{F}_b(M))$;

$\underline{\mathcal{F}_{leak}()}$
Ret $L_{leak}$;
```

When the output size of a RO is b, we write it by pub-RO_b. Dodis et al. showed that when SCF is the type-II scheme, $\mathsf{MD}^{\mathsf{SCF}^{\mathcal{C}_{d,n}}}$ is indifferentiable from pub-RO_n up to $\mathcal{O}(2^{n/2})$ query complexity.

3 Security of Encryption Schemes Using (KDF-)MD-typeII Hash Functions

We customize the indifferentiability framework [28] called "the indifferentiability with condition". By using the framework, we show that OAEP, RSA-KEM, PSEC-KEM, ECIES-KEM and many other encryption schemes using (KDF-)MD-typeII hash functions are secure in the ICM.

3.1 Indifferentiability with Condition

We propose indifferentiability with condition. In this framework, we restrict queries to a private interface by some condition. Let P be an ideal primitive and H^P be a hash function.

Definition 3. *H^P is (t_A, t_S, ϵ) indifferentiable from random oracle $\mathcal{F}_n$ under condition α, denoted $H^P \sqsubset_\alpha \mathcal{F}_n$, if there exists simulator S of running time at most t_S such that for any distinguisher A of running time at most t_A such that queries from A to $H^P/\mathcal{F}_n$ are restricted by condition α $|Pr[A^{H^P,P} \Rightarrow 1] - Pr[A^{\mathcal{F}_n,S(\mathcal{F}_n)} \Rightarrow 1] \leq \epsilon$.*

From the definition, the following theorem is obtained.

Theorem 1. *Let C be any cryptosystem wherein queries to hash functions are restricted to condition α. Then, $H^P \sqsubset_\alpha \mathcal{F}_n \Leftrightarrow C(H^P) \succ C(\mathcal{F}_n)$.*

Proof. Let us start with the first implication ("$\Rightarrow$"). Assume that $\forall A, \exists S$ such that A is restricted by the condition α, $|Pr[A^{H^P,P} \Rightarrow 1] - Pr[A^{\mathcal{F}_n,S(\mathcal{F}_n)} \Rightarrow 1]| \leq \epsilon$ and ϵ is neglitible. We show that $\forall Env, \forall \mathcal{A}, \exists \mathcal{A}' : |Pr[Env^{\mathcal{C}(H^P),\mathcal{A}} \Rightarrow 1] - Pr[Env^{\mathcal{C}(\mathcal{F}_n),\mathcal{A}'} \Rightarrow 1]| \leq \epsilon$ such that $\mathcal{C}$ satisfies condition α. Since for $\forall A$ $\exists S$ $|Pr[A^{H^P,P} \Rightarrow 1] - Pr[A^{\mathcal{F}_n,S(\mathcal{F}_n)} \Rightarrow 1]| \leq \epsilon$ holds such that A is restricted by the condition α, when $A = Env^{\mathcal{C},\mathcal{A}}$ where Env is any environment and $\mathcal{A}$ is any attacker for cryptosystem $\mathcal{C}$, $|Pr[A^{H^P,P} \Rightarrow 1] - Pr[A^{\mathcal{F}_n,S(\mathcal{F}_n)} \Rightarrow 1]| \leq \epsilon$ holds. We define attacker $\mathcal{A}'$ by combining $\mathcal{A}$ and S. Then, $|Pr[Env^{\mathcal{C}(H^P),\mathcal{A}} \Rightarrow 1] - Pr[Env^{\mathcal{C}(\mathcal{F}_n),\mathcal{A}'} \Rightarrow 1]| \leq \epsilon$ holds.

The second implication ("$\Leftarrow$") is proven by using the same as the proof of Theorem 1 of [28]. Since we do not use this result, we omit its proof. □

3.2 Indifferentiability Results for (KDF-)MD-type-II Hash Functions

First we pick up the condition "prefix-free". Since input sizes of OAEP, RSA-KEM, PSEC-KEM, ECIES-KEM and many other encryption schemes are fixed, these cryptosystems satisfy the condition "prefix-free".

Second we prove that (KDF-)MD-TypeII hash functions are indifferentiable from ROs under the condition "prefix-free". Let P be an ideal function, H^P be a hash function using P and G^P be a hash function H^P with a prefix-free padding pfpad. Namely $G^P(M) = H^P(\mathsf{pfpad}(M))$. First we show that if G^P is indifferentiable from RO, H^P is indifferentiable from RO under the condition "prefix-free" (Theorem 2).

Theorem 2. *$G^P \sqsubset \mathcal{F}_n \Rightarrow H^P \sqsubset_\alpha \mathcal{F}_n$ where α is the condition "prefix-free".*

Proof. We assume that $G^P \sqsubset \mathcal{F}_n$. Namely $|Pr[A^{\mathcal{F}_n,S} \Rightarrow 1] - Pr[A^{G^P,P} \Rightarrow 1]| \leq \epsilon$ and ϵ is negligible. We modify $\mathcal{F}_n$ to $\mathcal{F}_n \circ \mathsf{pfpad}$. Since pfpad is an injective function, for a fresh query $\mathcal{F}_n \circ \mathsf{pfpad}$ returns a freshly-chosen random value. Therefore, $|Pr[A^{\mathcal{F}_n,S} \Rightarrow 1] - Pr[A^{G^P,P} \Rightarrow 1]| \leq \epsilon \Rightarrow |Pr[A^{\mathcal{F}_n \circ \mathsf{pfpad},S} \Rightarrow 1] - Pr[A^{H^P \circ \mathsf{pfpad},P} \Rightarrow 1]| \leq \epsilon$. Note that $G^P = H^P \circ \mathsf{pfpad}$. We define a new distinguisher A_1 by combining A with pfpad. Thus $|Pr[A^{\mathcal{F}_n \circ \mathsf{pfpad},S} \Rightarrow 1] - Pr[A^{H^P \circ \mathsf{pfpad},P} \Rightarrow 1]| \leq \epsilon \Rightarrow |Pr[A_1^{\mathcal{F}_n,S} \Rightarrow 1] - Pr[A_1^{H^P,P} \Rightarrow 1]| \leq \epsilon$. Since A is any distinguisher and pfpad is any prefix-free padding, A_1 is any distinguisher where queries to $H^P/\mathcal{F}_n$ are restricted by condition "prefix-free". The proof is completed. □

By using Theorem 2, we show that (KDF-)MD-TypeII hash functions are indifferentiable from ROs under the condition "prefix-free". First we show that the MD-TypeII hash function with a prefix-free padding is indifferentiable from RO as follows.

Theorem 3. *Let* SCF *be the type-II scheme.* $\mathsf{PFMD}^{\mathsf{SCF}^{\mathcal{C}_{d,n}}} \sqsubset \mathcal{F}_n$ *where for any* t_A*,* $t_S = t_A + \mathcal{O}((q_E + q_D)^2)$

$$\epsilon \leq \frac{3(lq_H + q_E + q_D)^2 + (lq_H + q_E)^2 + 2(lq_H + q_E + q_D)}{2^{n+1}}$$

where A *can make queries to* $\mathsf{PFMD}^{\mathsf{SCF}^{\mathcal{C}_{d,n}}}/\mathcal{F}_n$ *at most* q_H *times where the maximum blocks of the query are* l *blocks and* A *can make queries to* E/S_E *and* D/S_D *at most* q_E *and* q_D *times, respectively.*

This proof is shown in Subsection 3.3.

For KDF1, we can see that $\mathcal{F}_n(*||\langle 0\rangle), \mathcal{F}_n(*||\langle 1\rangle), \ldots, \mathcal{F}_n(*||\langle m-1\rangle)$ are independent random oracles. A hash function concatenating m independent random oracles is a random oracle $\mathcal{F}_{mn}$. The same is true for KDF2 and KDF3. Thus, the following theorem holds.

Theorem 4. *For* $i = 1, 2,$ *and* 3 $\mathsf{KDF}i$*-*$\mathcal{F}_n \sqsubset \mathcal{F}_{mn}$ *where for any* t_A*,* $t_S = t_A + \mathcal{O}(q)$*, and* $\epsilon = 0$ *where* A *can make queries to* $\mathcal{F}_n/S$ *at most* q *times.*

Result. By combining above theorems, (KDF-)MD-typeII hash functions are indifferentiable from ROs. The indifferentiable result and Theorem 1 yield the result that OAEP, RSA-KEM, PSEC-KEM, ECIES-KEM and many other encryption schemes are secure when ROs are instantiated with (KDF-)MD-TypeII hash functions.

3.3 Proof of Theorem 3

We define a simulator $S = (S_E, S_D)$ as follows.

```
simulator $S_E(k, x)$
001 If $\mathsf{E}(k, x) \neq \perp$, ret $\mathsf{E}(k, x)$;
002 $(m, v) \leftarrow C^{-\mathrm{PRE}}(k, x)$;
003 $y \xleftarrow{\$} \{0, 1\}^n$;
004 $\mathsf{V}(IV) \leftarrow \varepsilon$;
005 If $\mathsf{V}(v) \neq \perp$,
006     If $\exists M$ s.t. $\mathsf{pfpad}(M) = \mathsf{V}(v)||m$,
007         $w \leftarrow \mathcal{F}_n(M)$;
008         $y \leftarrow C^{-\mathrm{POST}}(v, m, w)$;
009     $w \leftarrow C^{\mathrm{POST}}(v, m, y)$;
010     $\mathsf{V}(w) \leftarrow \mathsf{V}(v)||m$;
011 $\mathsf{E}(k, x) \leftarrow y$; $\mathsf{D}(k, y) \leftarrow x$;
012 Ret $y$;
```

```
simulator $S_D(k, y)$
101 If $\mathsf{D}(k, y) \neq \perp$, ret $\mathsf{D}(k, y)$
102 $x \xleftarrow{\$} \{0, 1\}^n$;
103 $\mathsf{V}(IV) \leftarrow \varepsilon$;
104 For all $v$ s.t. $\mathsf{V}(v) \neq \perp$
105     Calculate $x'$ and $m$ where
        $v = C_1^{-\mathrm{PRE}}(k, x')$ and $(v, m) = C^{-\mathrm{PRE}}(k, x')$;
106     If $\exists M$ s.t. $\mathsf{pfpad}(M) = \mathsf{V}(v)||m$,
107         $w \leftarrow \mathcal{F}_n(M)$;
108         If $w = C^{\mathrm{POST}}(v, m, y)$,
109             $x \leftarrow x'$ and goto Line 110;
110 $\mathsf{E}(k, x) \leftarrow y$; $\mathsf{D}(k, y) \leftarrow x$;
111 Ret $x$;
```

S_E simulates E and S_D simulates D. In the following proof, we write an input-output triple of S_E/E and S_D/D by (k, x, y), the input of the type-II scheme using S_E by (v, m) and the output by w. Namely $(v, m) \leftarrow C^{-\mathrm{PRE}}(k, x)$ and $w \leftarrow C^{\mathrm{POST}}(v, m, y)$. We define chain triples and pf-chain triples.

Definition 4. $(k_1, x_1, y_1), \ldots, (k_i, x_i, y_i)$ *are chain triples if* $v_1 = IV$*,* $w_t = v_{t+1}$ $(t = 1, \ldots, i-1)$ *and there does not exist* M *such that* $\mathsf{pfpad}(M) = m_1|| \cdots ||m_i$*.*

Definition 5. $(k_1, x_1, y_1), \ldots, (k_i, x_i, y_i)$ *are pf-chain triples if* $(k_1, x_1, y_1), \ldots, (k_{i-1}, x_{i-1}, y_{i-1})$ *are chain triples,* $w_{i-1} = v_i$ *and there exists* M *such that* $\mathsf{pfpad}(M) = m_1|| \cdots ||m_i$*.*

Tables E and D record all input-output triples of S_E and S_D. The table V records all messages for chain-triples.

We give a proof using the game sequences Game 0, Game 1, and Game 2. In this proof, A interacts $\mathcal{O}_H$, $\mathcal{O}_E$ and $\mathcal{O}_D$.

- **Game 0**: This game is the RO scenario. Namely, $\mathcal{O}_H = \mathcal{F}_n$, $\mathcal{O}_E = S_E$ and $\mathcal{O}_D = S_D$.
- **Game 1**: In this game, we modify $\mathcal{O}_H$ by $\mathcal{O}_H = \mathsf{PFMD}^{\mathsf{SCF}^S}$. Note that $\mathsf{PFMD}^{\mathsf{SCF}^S}$ is the PFMD hash function with the type-II scheme using S_E.
- **Game 2**: This is the final game. In this game, we modify all oracles; $\mathcal{O}_H = \mathsf{PFMD}^{\mathsf{SCF}^{\mathcal{C}_{d,n}}}$, $\mathcal{O}_E = E$ and $\mathcal{O}_D = D$. Namely, this game is the ideal cipher scenario.

Game 0→Game 1: We prove that Game 0 is equal to Game 1 unless the following bad events occur.

- Event $\mathsf{E1}$: The triple (k, x, y) is such that (k, x, y) is defined by $\mathcal{O}_E$ and there is another triple (k', x', y') such that $w = w'$ and (k', x', y') is defined by $\mathcal{O}_E$.
- Event $\mathsf{E2}$: The triple (k, x, y) is such that (k, x, y) is defined by $\mathcal{O}_E$ and $w = IV$.
- Event $\mathsf{E3}$: The triple (k, x, y) is such that (k, x, y) is defined by $\mathcal{O}_E$ and there is another triple (k', x', y') such that $w = v'$ and (k', x', y') is defined before (k, x, y) is defined.
- Event $\mathsf{E4}$: The triple (k, x, y) is such that (k, x, y) is defined by $\mathcal{O}_D$ and there exist triples $(k_1, x_1, y_1), \ldots, (k_i, x_i, y_i)$ such that $(k_1, x_1, y_1), \ldots, (k_i, x_i, y_i), (k, x, y)$ are chain triples.
- Event $\mathsf{E5}$: The triple (k, x, y) is such that (k, x, y) is defined by $\mathcal{O}_D$, $v = IV$ and there does not exists M such that $\mathsf{pfpad}(M) = m$.

In order to prove that Game 0 is equal to Game 1 unless the following bad events occur, we use the technique of [22]. Namely, we show the following three points.

1. In Game 0, unless a bad event occurs, the answers given by $\mathcal{O}_E$ and $\mathcal{O}_D$ are consistent with those given by $\mathcal{O}_H$.
2. In Game 1, unless a bad event occurs, the answers given by $\mathcal{O}_E$ and $\mathcal{O}_D$ are consistent with those given by $\mathcal{O}_H$.
3. Unless a bad event occurs, for any M $\mathcal{O}_H(M) = \mathcal{F}_n(M)$ in Game 0 and Game 1.

Let $G0$ and $G1$ be events that A outputs 1 in Game 0 and Game 1, respectively. If the above three points hold, $|Pr[G1] - Pr[G0]| \leq Pr[\mathsf{E1} \vee \mathsf{E2} \vee \mathsf{E3} \vee \mathsf{E4} \vee \mathsf{E5}] \leq Pr[\mathsf{E1}] + Pr[\mathsf{E2}] + Pr[\mathsf{E3}] + Pr[\mathsf{E4}] + Pr[\mathsf{E5}]$. So we show that $Pr[\mathsf{E1}], Pr[\mathsf{E2}], Pr[\mathsf{E3}], Pr[\mathsf{E4}]$ and $Pr[\mathsf{E5}]$ are negligible.

Before starting the proof of the above points, we give a useful lemma.

Lemma 2. *For any pf-chain triples* $(k_1, x_1, y_1), \ldots, (k_i, x_i, y_i)$*, unless a bad event occurs,* $w_i = \mathcal{F}_n(M^*)$ *where* $\mathsf{pfpad}(M^*) = m_1 || \cdots || m_i$.

Proof. To the contrary, assume that there are pf-chain triples $(k_1, x_1, y_1), \ldots, (k_i, x_i, y_i)$ such that $w_i \neq \mathcal{F}_n(M^*)$ where $\mathsf{pfpad}(M^*) = m_1 || \cdots || m_i$. We divide this case into the following cases.

1. (k_i, x_i, y_i) is defined by $\mathcal{O}_E$.
   (a) (k_i, x_i, y_i) is defined in line 003.
   (b) (k_i, x_i, y_i) is defined in line 008.
2. (k_i, x_i, y_i) is defined by $\mathcal{O}_D$.
   (a) (k_i, x_i, y_i) is defined in line 102.
   (b) (k_i, x_i, y_i) is defined in line 109.

Since if $i = 1$ $w_1 = \mathcal{F}_n(M^*)$ holds due to lines 005-010 and lines 104-109, we assume that $i > 1$.

First we consider the case 1-a. Since y_i is defined in line 003 ($\mathsf{V}(v_{i-1}) =\perp$), when (k_i, x_i, y_i) is defined, some triple (k_t, x_t, y_t) in triples $(k_1, x_1, y_1), \ldots, (k_{i-1}, x_{i-1}, y_{i-1})$ is not defined. We assume that t is the maximum value such that (k_t, x_t, y_t) is defined after (k_i, x_i, y_i) is defined.

- Case $t = 1$: If (k_1, x_1, y_1) is defined by the $\mathcal{O}_E$ query, since (k_1, x_1, y_1) is defined after (k_2, x_2, y_2) is defined and $w_1 = v_2$, event $\mathsf{E3}$ occurs. If (k_1, x_1, y_1) is defined by the $\mathcal{O}_D$ query, since pfpad is a prefix-free padding and m_1 is the prefix of $m_1 || \cdots || m_i$, there does not exist M such that $\mathsf{pfpad}(M) = m_1$. Since $v_1 = IV$, event $\mathsf{E5}$ occurs.
- Case $1 < t < i$: If (k_t, x_t, y_t) is defined by the $\mathcal{O}_E$ query, since $w_t = v_{t+1}$ and $(k_{t+1}, x_{t+1}, y_{t+1})$ is defined before (k_t, x_t, y_t) is defined, event $\mathsf{E3}$ occurs. If (k_t, x_t, y_t) is defined by the $\mathcal{O}_D$ query, since $w_{t-1} = v_t$, $(k_1, x_1, y_1), \ldots, (k_t, x_t, y_t)$ are chain triples and (k_t, x_t, y_t) is defined before $(k_{t-1}, x_{t-1}, y_{t-1})$ is defined, event $\mathsf{E4}$ occurs.

We consider the case 1-b. In this case, since $\mathsf{V}(v_{i-1}) \neq\perp$, when (k_i, x_i, y_i) is defined, there exists M such that $\mathsf{pfpad}(M) = \mathsf{V}(v_{i-1})||m_i$ and $w_i = \mathcal{F}_n(M)$. Since $w_i \neq \mathcal{F}_n(M^*)$, $\mathsf{V}(v_{i-1}) \neq m_1 || \cdots || m_{i-1}$. Namely, there are another chain triples $(k'_1, x'_1, y'_1), \ldots, (k'_j, x'_j, y'_j)$ such that $(k'_1, x'_1, y'_1), \ldots, (k'_j, x'_j, y'_j), (k_1, x_i, y_i)$ are pf-chain triples where $\mathsf{V}(v_{i-1}) = m'_1 || \cdots || m'_j$. We divide the case into the following cases.

- $(k'_1, x'_1, y'_1), \ldots, (k'_j, x'_j, y'_j), (k_1, x_1, y_1), \ldots, (k_{i-1}, x_{i-1}, y_{i-1})$ are defined by $\mathcal{O}_E$: Since $w_{i-1} = w'_j$ and $m'_1 || \cdots || m'_j \neq m_1 || \cdots || m_{i-1}$, a collision occurs for the hash function iterating the type-II scheme using $\mathcal{O}_E$. Since a collision of the hash function can be reduced into an event of the compression function; finding a collision or finding a preimage of IV, event $\mathsf{E1}$ or $\mathsf{E2}$ occurs.
- Some triple (k_t, x_t, y_t) of $(k'_1, x'_1, y'_1), \ldots, (k'_j, x'_j, y'_j), (k_1, x_1, y_1), \ldots, (k_{i-1}, x_{i-1}, y_{i-1})$ is defined by $\mathcal{O}_D$: We assume that t is the minimum value. When $t = 1$, $\mathsf{E5}$ occurs from the same discussion as the case 1-a-($t = 1$). When $t > 2$, if $(k_{t-1}, x_{t-1}, y_{t-1})$ is defined after (k_t, x_t, y_t) is defined, since $(k_{t-1}, x_{t-1}, y_{t-1})$ is defined by $\mathcal{O}_E$ (t is the minimum value), event $\mathsf{E3}$ occurs. If (k_t, x_t, y_t) is defined after $(k_{t-1}, x_{t-1}, y_{t-1})$ is defined, event $\mathsf{E4}$ occurs from the same discussion as the case 1-a-($1 < t < i$).

We consider the case 2-a. Since y_i is defined in line 102 ($\mathsf{V}(v_{i-1}) =\perp$), when (k_i, x_i, y_i) is defined, there does not exist some triple (k_t, x_t, y_t) such that $t < i$ and (k_t, x_t, y_t) is defined after (k_i, x_i, y_i) is defined. This case is equal to the case 1-a. Therefore, in this case event $\mathsf{E3}, \mathsf{E4}$ or $\mathsf{E5}$ occurs.

Finally we consider the case 2-b. In this case, since $\mathsf{V}(v_{i-1}) \neq\perp$, when (k_i, x_i, y_i) is defined, there exists M such that $\mathsf{pfpad}(M) = \mathsf{V}(v_{i-1})||m_i$, $w_i = \mathcal{F}_n(M)$ and $\mathsf{V}(v_{i-1}) \neq m_1 || \cdots || m_{i-1}$. This case is equal to the case 1-b. Therefore, in this case event $\mathsf{E1}, \mathsf{E2}, \mathsf{E3}, \mathsf{E4}$ or $\mathsf{E5}$ occurs.

The proof of the lemma is completed. □

By using the lemma, we prove the three points.

Proof of point 1. From Lemma 2, for any pf-chain triples $(k_1, x_1, y_1), \ldots, (k_i, x_i, y_i)$, unless a bad event occurs, $w_i = \mathcal{F}_n(M^*)$ where $\mathsf{pfpad}(M^*) = m_1 || \cdots || m_i$. Since $\mathcal{O}_H = \mathcal{F}_n$, the answers given by $\mathcal{O}_E$ and $\mathcal{O}_D$ are consistent with those given by $\mathcal{O}_H$.

Proof of point 2. Since $\mathcal{O}_H$ uses $\mathcal{O}_E$ ($\mathcal{O}_H = \mathsf{PFMD}^{\mathsf{SCF}^S}$), the answers given by $\mathcal{O}_E$ and $\mathcal{O}_D$ are consistent with those given by $\mathcal{O}_H$.

Proof of point 3. From Lemma 2, unless a bad event occurs, in Game 1 for any M $\mathcal{O}_H(M) = \mathcal{F}_n(M)$. And in Game 0 $\mathcal{O}_H = \mathcal{F}_n$.

Thus Game 1 is equal to Game 0 unless a bad event occurs.

Next we evaluate the probabilities $Pr[\mathsf{E1}], Pr[\mathsf{E2}], Pr[\mathsf{E3}], Pr[\mathsf{E4}]$ and $Pr[\mathsf{E5}]$.

- $Pr[\mathsf{E1}]$: This is the collision event for SCF^S. Since an output of S_E is chosen uniformly from $\{0,1\}^n$ and $C^{\mathrm{POST}}(v,m,\cdot)$ is bijective, for any triples $(k,x,y),(k',x',y')$, w and w' are chosen uniformly from $\{0,1\}^n$. Since the maximum number of times that S_E is called is $lq_H + q_E$, from the birthday analysis $Pr[\mathsf{E1}] \leq \frac{(lq_H+q_E)^2}{2^{n+1}}$.
- $Pr[\mathsf{E2}]$: This is the event of finding a preimage of IV for SCF^S. Since the maximum number of times that S_E is called is $lq_H + q_E$, $Pr[\mathsf{E2}] \leq \frac{lq_H+q_E}{2^n}$.
- $Pr[\mathsf{E3}]$: Since y is chosen uniformly from $\{0,1\}^n$ that is independent from (k',x',y'), w are chosen uniformly from $\{0,1\}^n$ and independently from (k',x',y'). Since the maximum number of such triple is $lq_H + q_E + q_D$, $Pr[\mathsf{E3}] \leq \frac{(lq_H+q_E+q_D)(lq_H+q_E)}{2^n}$.
- $Pr[\mathsf{E4}]$: Since $(k_1,x_1,y_1),\ldots,(k_i,x_i,y_i),(k,x,y)$ are chain triples (not pf-chain triples), x is chosen uniformly from $\{0,1\}^n$ and independently from (k_i,x_i,y_i). Since $C_1^{-\mathrm{PRE}}(k,\cdot)$ is bijective, v is chosen uniformly from $\{0,1\}^n$. Thus, since S_D is called at most q_D times and the maximum number of triple (k_i,x_i,y_i) is $lq_H + q_E + q_D$, $Pr[\mathsf{E4}] \leq \frac{q_D(lq_H+q_E+q_D)}{2^n}$.
- $Pr[\mathsf{E5}]$: Since there does not exist M such that $\mathsf{pfpad}(M) = m$, triple (k,x,y) is defined in line 102. Since $C_1^{-\mathrm{PRE}}(k,\cdot)$ is bijective, v is chosen uniformly from $\{0,1\}^n$. Thus $Pr[\mathsf{E5}] \leq \frac{q_D}{2^n}$

Thus $|Pr[G1] - Pr[G0]| \leq \frac{2(lq_H+q_E+q_D)^2+(lq_H+q_E)^2+2(lq_H+q_E+q_D)}{2^{n+1}}$.

Game 1→Game 2: Let $G2$ be an event that A outputs 1 in Game 2. Since outputs of S_E and S_D are chosen uniformly from $\{0,1\}^n$, $S_E = E$ and $S_D = D$ unless a collision occurs. Thus we have via a straightforward birthday analysis that $|Pr[G2] - Pr[G1]| \leq \frac{(lq_H+q_E+q_D)^2}{2^{n+1}}$.

The proof of the theorem is completed. □

4 Security of Pub-RO Secure Cryptosystems Using KDF-MD-typeII Hash Functions

In this section, by using the WRO approach, we show that pub-RO secure cryptosystems are secure when ROs are instantiated with the KDF-MD-TypeII hash functions. Note that pub-RO secure cryptosystems are that all inputs of hash functions are public (e.g. FDH, PFDH, Fiat-Shamir, BLS, PSS, a variant of Boneh-Franklin IBE and Boneh-Boyern IBE). First we show that KDFs using pub-RO are differentiable from pub-RO. Therefore, we cannot trivially extend the result of [19] to a proof of indifferentiability for KDF-MD-TypeII hash functions. Therefore, we propose a new WRO, called private interface leaking random oracle (privleak-RO). Roughly speaking, privleak-RO leaks all input-output pairs of the private interface of RO but does not leak an input-output pairs of the public interface of RO. Since an adversary can obtain all inputs of hash functions in pub-RO secure cryptosystems in the RO model, the pub-RO secure cryptosystems are secure in the privleak-RO model. We show that KDFs using pub-RO are indifferentiable from privleak-ROs. Since the MD-TypeII hash function is indifferentiable from pub-RO [19], the KDF-MD-TypeII hash functions are indifferentiable from privleak-ROs.

4.1 Differentiable Attack for KDFs using pub-RO

We show that $\mathsf{KDF1\text{-}pub\text{-}}\mathrm{RO}_n$, $\mathsf{KDF2\text{-}pub\text{-}}\mathrm{RO}_n$ and $\mathsf{KDF3\text{-}pub\text{-}}\mathrm{RO}_n$ are differentiable from $\mathsf{pub\text{-}}\mathrm{RO}_{nm}$ as follows. We only show that $\mathsf{KDF1\text{-}pub\text{-}}\mathrm{RO}_n$ is differentiable from $\mathsf{pub\text{-}}\mathrm{RO}_{nm}$. For $\mathsf{KDF2\text{-}pub\text{-}}\mathrm{RO}_n$ and $\mathsf{KDF3\text{-}pub\text{-}}\mathrm{RO}_n$, we can prove them by the similar proof.

Let $S = (S_{leak}, S_{\mathcal{F}^{pub}})$ be any simulator that simulates $\mathcal{F}_{leak}$ and $\mathcal{F}^{pub}_{nm}$ respectively. Let $\mathcal{O}_H = \mathsf{KDF1\text{-}}\mathcal{F}_n/\mathcal{F}_{nm}$, $\mathcal{O}_{leak} = \mathcal{F}_{leak}/S_{leak}$ and $\mathcal{O}_{\mathcal{F}} = \mathcal{F}^{pub}_n/S_{\mathcal{F}^{pub}}$. We define a distinguisher A as follows.

1. For $i = 1, \dots, q_H/2$ (where q_H is the maximum number of queries to $\mathcal{O}_H$ made by distinguisher A)
   (a) $j \xleftarrow{\$} \{0,1\}$;
   (b) $M \xleftarrow{\$} \{0,1\}^{ns}$ such that $1 \le sn \le l$ where l is the maximum bit length of a $\mathcal{O}_H$ query;
   (c) Makes a query $M||\langle 0\rangle$ to $\mathcal{O}_{\mathcal{F}^{pub}}$ and receives w;
   (d) If $j = 0$, makes a query M to $\mathcal{O}_H$;
   (e) Makes a query to $\mathcal{O}_{leak}$ and receives list L;
   (f) Makes a query M to $\mathcal{O}_H$ and receives z;
   (g) If $z[1] \neq w$, return 1;
   (h) If $j = 0$ and there does not exist $(M||\langle 1\rangle, z[2])$ in L, return 1;
2. return 0;

Consider that A interacts with $(\mathcal{F}^{priv}_{nm}, S)$. When S does not make query M to $\mathcal{F}_{nm}(M)$, the probability that A returns 0 is negligible due to step 1-g. This implies that S should make the query M to $\mathcal{F}_{nm}(M)$. Thus when the step 1-e is executed, in list L_{leak} of $\mathcal{F}_{nm}$ the pair (M, z) shold be stored. When step 1-e is invoked, S does not know whether A makes query M to $\mathcal{F}^{priv}_{mn}$ or not. Note that if $j = 0$, pairs $(M||\langle 0\rangle, z[1]), (M||\langle 1\rangle, z[2]), \dots$ should be stored in list L and if $j = 1$, only the pair $(M||\langle 0\rangle, z[1])$ should be stored in list L. Since j is chosen uniformly from $\{0,1\}$, when A interacts with $(\mathcal{F}^{priv}_{nm}, S)$, in Step 1-e S_{leak} mistakes the simulation, thus A outputs 1 with non-negligible probability. On the other hand, when A interacts with $(\mathsf{KDF1\text{-}}\mathcal{F}^{priv}_n, \mathcal{F}^{pub}_n, \mathcal{F}_{leak})$, A explicitly outputs 0 with probability of 1. Therefore, $\mathsf{KDF1\text{-}pub\text{-}}\mathrm{RO}_n$ is differentiable from $\mathsf{pub\text{-}}\mathrm{RO}_{nm}$. We can prove that $\mathsf{KDF2\text{-}pub\text{-}}\mathrm{RO}_n$ and $\mathsf{KDF3\text{-}pub\text{-}}\mathrm{RO}_n$ are differentiable from $\mathsf{pub\text{-}}\mathrm{RO}_{nm}$s by using the same as the above attack. To avoid the attack, we define the privleak-RO to avoid the above attack.

4.2 privleak-RO

Since no simulator can know whether a pair in L_{leak} is defined on the public interface or the private interface, the above attack works. So we define privleak-RO so that S can know all input-output pair defined on the private interface. Privleak-RO consists of a random oracle $\mathcal{F}_b$ and a private interface leak oracle $\mathcal{F}_{privleak}$ where the output size of $\mathcal{F}_b$ is b bits. Let $\mathcal{F}^{priv}_b$ be a private interface of a RO and $\mathcal{F}^{pub}_b$ be a public interface of a RO. $\mathcal{F}_{privleak}$ leaks all input-output pairs of $\mathcal{F}^{priv}_b$. The description is as follows.

```
$\underline{\mathcal{F}_b^{priv}(M)}$
If $\mathsf{F}_b(M) \neq \perp$,
    $L_{privleak} \leftarrow (M, \mathsf{F}_b(M))$;
    Ret $\mathsf{F}_b(M)$;
$\mathsf{F}_b(M) \xleftarrow{\$} \{0,1\}^b$;
$L_{privleak} \leftarrow (M, \mathsf{F}_b(M))$;
Ret $\mathsf{F}_b(M)$;
```

```
$\underline{\mathcal{F}_b^{pub}(M)}$
If $\mathsf{F}_b(M) \neq \perp$, ret $\mathsf{F}_b(M)$;
$\mathsf{F}_b(M) \xleftarrow{\$} \{0,1\}^b$;
Ret $\mathsf{F}_b(M)$;
```

```
$\underline{\mathcal{F}_{privleak}()}$
Ret $L_{privleak}$;
```

When the output size of a RO is b-bits, we denote it by privleak-RO$_b$.

4.3 Indifferentiability Results for KDFs

We show that KDFs using pub-RO$_n$ are indifferentiable from privleak-RO$_{nm}$ as follows.

Theorem 5. $\mathsf{KDF}i\text{-}\mathsf{pub}$-*$RO_n \sqsubset$ privleak-RO ($i = 1, 2, 3$) where for any t_A, $t_S = t_A + \mathcal{O}(q_E + q_D)$ and $\epsilon = 0$.*

We give the proof of $\mathsf{KDF1}$-pub-RO$_n$ $\sqsubset$ privleak-RO in Subsection 4.4. We can prove that $\mathsf{KDF2}$-pub-RO$_n$ $\sqsubset$ privleak-RO and $\mathsf{KDF3}$-pub-RO$_n$ $\sqsubset$ privleak-RO by the same as the proof of Theorem 5. So we ommit these proofs.

Result. From Theorem 5 and Theorems 7.1 and 7.2 of [19], $\mathsf{KDF}i\text{-}\mathsf{MD}^{\mathsf{SCF}^{\mathcal{C}_{d,n}}} \sqsubset$ privleak-RO ($i = 1, 2, 3$) hold. Since pub-RO secure cryptosystems are secure in the privleak-RO model, these cryptosystems are secure when ROs are instantiated with KDF-MD-typeII Hash Functions.

4.4 Proof of Theorem 5

Recall the experiment of this proof. A interacts with ($\mathsf{KDF1}$-pub-RO$_n, \mathcal{F}_n, \mathcal{F}_{leak}$) or $(\mathcal{F}_{mn}^{priv}, S_{\mathcal{F}_n}, S_{leak})$ where $S_{\mathcal{F}_n}$ is a simulator of $\mathcal{F}_n^{pub}$ and S_{leak} is a simulator of $\mathcal{F}_{leak}$. We define a simulator $S = (S_{\mathcal{F}_n}, S_{leak})$ that simulates $\mathcal{F}_n$ and $\mathcal{F}_{leak}$ as follows. Let F_S be a (initially everywhere $\perp$) table that stores all input-output values of $S_{\mathcal{F}_n}$.

```
$\underline{S_{\mathcal{F}_n}(M)}$
If $\mathsf{F}_S(M) \neq \perp$, ret $\mathsf{F}_S(M)$;
If $M_{[64]} = \langle t \rangle \in \{\langle 0 \rangle, \ldots, \langle m-1 \rangle\}$,
    $w \leftarrow \mathcal{F}_{mn}^{pub}(M_{(64)})$;
    $\mathsf{F}_S(M) \leftarrow w[t+1]$;
Else $\mathsf{F}_S(M) \leftarrow \{0,1\}^n$;
$L_S \leftarrow (M, \mathsf{F}_S(M))$;
Ret $\mathsf{F}_S(M)$;
```

```
$\underline{S_{privleak}()}$
$(M^1, w^1), \ldots, (M^j, w^j) \leftarrow \mathcal{F}_{leak}()$;
For $i = 1, \ldots, j$ and $t = 1, \ldots, m$,
    $L_S \leftarrow (M^i || \langle t-1 \rangle, w^i[t])$;
Ret $L_S$;
```

Since a last 64 bit value of an input of $\mathcal{F}_n$ in $\mathsf{KDF1}$-$\mathcal{F}_n$ is one of $\{\langle 0 \rangle, \ldots, \langle m-1 \rangle\}$, on a query x where $x_{[64]} \in \{\langle 0 \rangle, \ldots, \langle m-1 \rangle\}$ the output is defined by using $\mathcal{F}_{mn}^{pub}$ and on other type queries the outputs are defined by a random choice. We define $\mathcal{F}_{leak}$ such that it leaks input-output pairs of $S_{\mathcal{F}_n}$ and pairs that are defined by using $\mathcal{F}_{leak}$.

We give a proof using the game sequences Game 0, Game 1, ..., Game 6 that are shown in Figs. 1, 2, 3, 4, 5 and 6. Without loss of generality, we assume that distinguisher A does not

```
$\underline{\mathcal{O}_0(M)}$
For $j = 0, \ldots, m-1$
    $w_j \leftarrow$ choose-$\mathcal{F}_n(M||\langle j \rangle)$;
Ret $w_0|| \cdots ||w_{m-1}$;
$\underline{\mathcal{O}_1()}$
Ret $L_S$;
$\underline{\mathcal{O}_2(X)}$
Ret choose-$\mathcal{F}_n(X)$;

$\underline{\text{choose-}\mathcal{F}_n(X)}$
If $\mathsf{F}(X) = \perp$, $\mathsf{F}(X) \xleftarrow{\$} \{0,1\}^n$;
$L_S \leftarrow (X, \mathsf{F}(X))$;
Ret $\mathsf{F}(X)$;
```

Fig. 1. Game 0

```
$\underline{\mathcal{O}_0(M)}$
For $j = 0, \ldots, m-1$
    $w_j \leftarrow$ choose-$\mathcal{F}_n(M||\langle j \rangle)$;
Ret $w_0|| \cdots ||w_{m-1}$;
$\underline{\mathcal{O}_1()}$
Ret $L_S$;
$\underline{\mathcal{O}_2(X)}$
Ret choose-$\mathcal{F}_n(X)$;

$\underline{\text{choose-}\mathcal{F}_n(X)}$
If $X_{[64]} = \langle t \rangle \in \{0, \ldots, m-1\}$,
    For $j = 0, \ldots, m-1$
        If $\mathsf{F}_j(X_{(64)}) = \perp$, $\mathsf{F}_j(X_{(64)}) \xleftarrow{\$} \{0,1\}^n$;
    $w \leftarrow \mathsf{F}_t(X_{(64)})$;
Else
    If $\mathsf{F}(X) = \perp$, $\mathsf{F}(X) \xleftarrow{\$} \{0,1\}^n$;
    $w \leftarrow \mathsf{F}(X)$
$L_S \leftarrow (X, w)$;
Ret $w$;
```

Fig. 2. Game 1

```
$\underline{\mathcal{O}_0(M)}$
For $j = 0, \ldots, m-1$
    $w_j \leftarrow$ choose-$\mathcal{F}_n(M||\langle j \rangle)$;
Ret $w_0|| \cdots ||w_{m-1}$;
$\underline{\mathcal{O}_1()}$
Ret $L_S$;
$\underline{\mathcal{O}_2(X)}$
Ret choose-$\mathcal{F}_n(X)$;

$\underline{\text{choose-}\mathcal{F}_n(X)}$
If $X_{[64]} = \langle t \rangle \in \{0, \ldots, m-1\}$,
    If $\mathsf{F}^*(X_{(64)}) = \perp$, $\mathsf{F}^*(X_{(64)}) \xleftarrow{\$} \{0,1\}^{mn}$;
    $\boxed{\text{For } j = 0, \ldots, m-1}$
        $\boxed{\text{If } \mathsf{F}_j(X_{(64)}) = \perp,\ \mathsf{F}_j(X_{(64)}) \leftarrow \mathsf{F}^*(X_{(64)})[j+1];}$
    $w \leftarrow \mathsf{F}^*(X_{(64)})[t+1]$;
Else
    If $\mathsf{F}(X) = \perp$, $\mathsf{F}(X) \xleftarrow{\$} \{0,1\}^n$;
    $w \leftarrow \mathsf{F}(X)$;
$L_S \leftarrow (X, w)$; Ret $w$;
```

Fig. 3. Game 2 and Game 3

```
$\underline{\mathcal{O}_0(M)}$
For $j = 0, \ldots, m-1$
    $w_j \leftarrow$ choose-$\mathcal{F}_n(0, M||\langle j \rangle)$;
Ret $w_0|| \cdots ||w_{m-1}$;
$\underline{\mathcal{O}_1()}$
Ret $L_S$;
$\underline{\mathcal{O}_2(X)}$
If $X_{[64]} = \langle t \rangle \in \{\langle 0 \rangle, \ldots, \langle m-1 \rangle\}$,
    Ret choose-$\mathcal{F}_n(1, X)$;
Else ret choose-$\mathcal{F}_n(2, X)$;

$\underline{\text{choose-}\mathcal{F}_n(s, X)}$
If $s \neq 2$,
    If $\mathsf{F}^*(X_{(64)}) = \perp$, $\mathsf{F}^*(X_{(64)}) \xleftarrow{\$} \{0,1\}^{mn}$;
  $w \leftarrow \mathsf{F}^*(X_{(64)})[t+1]$; //$\langle t \rangle = X_{[64]}$
Else
    If $\mathsf{F}(X) = \perp$, $\mathsf{F}(X) \xleftarrow{\$} \{0,1\}^n$;
    $w \leftarrow \mathsf{F}(X)$
$L_S \leftarrow (X, w)$;
Ret $w$;
```

Fig. 4. Game 4

repeat a query to any of its oracles. In each game, A interacts with oracles $\mathcal{O}_0, \mathcal{O}_1$, and $\mathcal{O}_2$. Let Gi be the event that A outputs 1 in Game i.

```
$\underline{\mathcal{O}_0(M)}$
601 If $\mathsf{F}^*(M) = \perp$, $\mathsf{F}^*(M) \xleftarrow{\$} \{0,1\}^{mn}$;
602 For $j = 0, \ldots, m-1$,
603     $L_S \leftarrow (M||\langle j \rangle, \mathsf{F}^*[j+1])$;
604 Ret $\mathsf{F}^*(M)$;
$\underline{\mathcal{O}_1()}$
611 Ret $L_S$;
$\underline{\mathcal{O}_2(X)}$
621 If $X_{[64]} = \langle t \rangle \in \{\langle 0 \rangle, \ldots, \langle m-1 \rangle\}$,
622     Ret choose-$\mathcal{F}_n(1, X)$;
623 Else ret choose-$\mathcal{F}_n(2, X)$;

$\underline{\text{choose-}\mathcal{F}_n(s, X)}$
631 If $s = 1$,
632     If $\mathsf{F}^*(X_{(64)}) = \perp$, $\mathsf{F}^*(X_{(64)}) \xleftarrow{\$} \{0,1\}^{mn}$;
633     $w \leftarrow \mathsf{F}^*(X_{(64)})[t+1]$; $//\langle t \rangle = X_{[64]}$
634 If $s = 2$,
635     If $\mathsf{F}(X) = \perp$, $\mathsf{F}(X) \xleftarrow{\$} \{0,1\}^n$;
636     $w \leftarrow \mathsf{F}(X)$
637 $L_S \leftarrow (X, w)$;
638 Ret $w$;
```

Fig. 5. Game 5

```
$\underline{\mathcal{O}_0(M)}$
701 If $\mathsf{F}^*(M) = \perp$, $\mathsf{F}^*(M) \xleftarrow{\$} \{0,1\}^{mn}$;
702 $T \leftarrow (M, \mathsf{F}^*(M))$;
703 Ret $\mathsf{F}^*(M)$;
$\underline{\mathcal{O}_1()}$
711 $(M^1, w^1), \ldots, (M^i, w^i) \leftarrow T$;
712 For $j = 1, \ldots, i$ $t = 0, \ldots, m-1$,
713     $L_S \leftarrow (M^j||\langle t \rangle, w^j[t+1])$;
714 Ret $L_S$;
$\underline{\mathcal{O}_2(X)}$
721 If $X_{[64]} = \langle t \rangle \in \{\langle 0 \rangle, \ldots, \langle m-1 \rangle\}$,
722     Ret choose-$\mathcal{F}_n(1, X)$;
723 Else ret choose-$\mathcal{F}_n(2, X)$;

$\underline{\text{choose-}\mathcal{F}_n(s, X)}$
731 If $s = 1$,
732     If $\mathsf{F}^*(X_{(64)}) = \perp$, $\mathsf{F}^*(X_{(64)}) \xleftarrow{\$} \{0,1\}^{mn}$;
733     $w \leftarrow \mathsf{F}^*(X_{(64)})[t+1]$; $//\langle t \rangle = X_{[64]}$
734 If $s = 2$,
735     If $\mathsf{F}(X) = \perp$, $\mathsf{F}(X) \xleftarrow{\$} \{0,1\}^n$;
736     $w \leftarrow \mathsf{F}(X)$
737 $L_S \leftarrow (X, w)$;
738 Ret $w$;
```

Fig. 6. Game 6

Game 0. Choose-$\mathcal{F}_n$ behaves like a random oracle. In this oracle, (initially everywhere $\perp$) table F and (initially emply) list L_s are used. $\mathcal{O}_0$ is $\mathsf{KDF1}$-$\mathcal{F}_n$, $\mathcal{O}_1$ is $\mathcal{F}_{leak}$ and $\mathcal{O}_2$ is $\mathcal{F}_n$. This game is the pub-RO scenario and thus $Pr[G0] = Pr[A^{\mathsf{KDF1}\text{-}\mathcal{F}_n, \mathcal{F}_n, \mathcal{F}_{leak}} \Rightarrow 1]$.

Game 1. In this game, we modify the subroutine choose-$\mathcal{F}_n$. We use new (initially everywhere $\perp$) tables F_j $(j = 0, \ldots, m-1)$ in addition to table F. These tables are used and table F is not used if $X_{[64]} \in \{\langle 0 \rangle, \ldots, \langle m-1 \rangle\}$. Thus this modification explicitly does not affect the view of the distinguisher A and $Pr[G0] = Pr[G1]$.

Game 2 (boxed procedures included). In this game, we modify the procedure of the case of $X_{[64]} \in \{\langle 0 \rangle, \ldots, \langle m-1 \rangle\}$ in the subroutine choose-$\mathcal{F}_n$. $\mathsf{F}_1(X_{(64)}), \ldots, \mathsf{F}_m(X_{(64)})$ is defined in line 432 in advance. These values are stored in a new (initially everywhere $\perp$) table F^*. In line 435 an output is defined by $\mathsf{F}^*(X_{(64)})[t+1]$. Since $\mathsf{F}^*(X_{(64)})[t+1] = \mathsf{F}_t(X_{(64)})$, these modifications does not affect the view of the distinguisher A. Thus $Pr[G1] = Pr[G2]$.

Game 3 (boxed procedures removed). In this game, we remove boxed procedures (line 433 and line 434). Since tables $\mathsf{F}_0, \ldots, \mathsf{F}_{m-1}$ are not used in other procedures, this modification does not affect the view of A. Thus $Pr[G2] = Pr[G3]$.

Game 4. In this game, we modify $\mathcal{O}_2$ and choose-$\mathcal{F}_n$. Inputs of choose-$\mathcal{F}_n$ are two values. The first value s is such that $s = 0$ if choose-$\mathcal{F}_n$ is called in $\mathcal{O}_0$, $s = 1$ if choose-$\mathcal{F}_n$ is called in $\mathcal{O}_2$ and $X_{[64]} \in \{\langle 0\rangle, \ldots, \langle m-1\rangle\}$, and $s = 2$ if choose-$\mathcal{F}_n$ is called in $\mathcal{O}_2$ and $X_{[64]} \notin \{\langle 0\rangle, \ldots, \langle m-1\rangle\}$. Since when $s = 0$ or $s = 1$ $X_{[64]} \in \{\langle 0\rangle, \ldots, \langle m-1\rangle\}$ and Lines 531-533 are executed, these modifications do not affect the view of A. Thus $Pr[G3] = Pr[G4]$.

Game 5. In this game, we hard-code choose-$\mathcal{F}_n$ in lines 602-603 in $\mathcal{O}_0$ and remove the case of $s = 0$ in choose-$\mathcal{F}_n$. This modification does not affect the view of A. Thus $Pr[G4] = Pr[G5]$.

Game 6. This is the final game. We modify $\mathcal{O}_0$ and $\mathcal{O}_1$. We remove line 602-603 and all input-output pairs are stored in a new table T. Lines 602-603 is moved in lines 712-713. Since A cannot see these procedures, these modifications don't affect the view of A. Thus $Pr[G5] = Pr[G6]$.

In Game 6, $\mathcal{O}_0$ is equal to $\mathcal{F}_{mn}^{priv}$. $\mathcal{O}_1$ is equal to S_{leak}. $\mathcal{F}_{mn}^{pub}$ is hard-coded in lines 732-733. Thus $\mathcal{O}_2$ is equal to $S_{\mathcal{F}_n}$ and $Pr[G6] = Pr[A^{\mathcal{F}_{mn}^{priv}, S}]$. The proof is completed. □

References

1. Masayuki Abe, Eike Kiltz, and Tatsuaki Okamoto. Chosen-Ciphertext Security with Optimal Ciphertext Overhead. In *ASIACRYPT*, volume 5350 of *Lecture Notes in Computer Science*, pages 355–371. Springer, 2008.
2. Mihir Bellare and Phillip Rogaway. Random Oracles are Practical: A Paradigm for Designing Efficient Protocols. In *ACM Conference on Computer and Communications Security*, pages 62–73, 1993.
3. Mihir Bellare and Phillip Rogaway. Optimal Asymmetric Encryption. In *EUROCRYPT*, volume 950 of *Lecture Notes in Computer Science*, pages 92–111. Springer, 1994.
4. Mihir Bellare and Phillip Rogaway. The Exact Security of Digital Signatures - How to Sign with RSA and Rabin. In *EUROCRYPT*, volume 1070 of *Lecture Notes in Computer Science*, pages 399–416. Springer, 1996.
5. John Black, Phillip Rogaway, and Thomas Shrimpton. Black-Box Analysis of the Block-Cipher-Based Hash-Functions Constructions from PGV. In *CRYPTO*, volume 2442 of *Lecture Notes in Computer Science*, pages 103–118. Springer, 2002.
6. Alexandra Boldyreva and Marc Fischlin. Analysis of random oracle instantiation scenarios for OAEP and other practical schemes. In *CRYPTO*, pages 412–429, 2005.
7. Dan Boneh. Simplified OAEP for the RSA and Rabin functions. In *CRYPTO*, volume 2139 of *Lecture Notes in Computer Science*, pages 275–291. Springer, 2001.
8. Dan Boneh and Xavier Boyen. Short signatures from the weil pairing. In *ASIACRYPT*, pages 514–532, 2001.
9. Dan Boneh and Xavier Boyen. Efficient selective-id secure identity-based encryption without random oracles. In *EUROCRYPT*, pages 223–238, 2004.
10. Daniel R. L. Brown. What hashes make RSA-OAEP secure? Cryptology ePrint Archive, Report 2006/223. 2006.
11. Ran Canetti, Oded Goldreich, and Shai Halevi. The Random Oracle Methodology, Revisited (Preliminary Version). In *STOC*, pages 209–218, 1998.
12. Donghoon Chang, Sangjin Lee, Mridul Nandi, and Moti Yung. Indifferentiable Security Analysis of Popular Hash Functions with Prefix-Free Padding. In *ASIACRYPT*, volume 4284 of *Lecture Notes in Computer Science*, pages 283–298. Springer, 2006.
13. Benoît Chevallier-Mames, Duong Hieu Phan, and David Pointcheval. Optimal Asymmetric Encryption and Signature Paddings. In *ACNS*, pages 254–268, 2005.
14. Jean-Sébastien Coron. Optimal Security Proofs for PSS and Other Signature Schemes. In *EUROCRYPT*, volume 2332 of *Lecture Notes in Computer Science*, pages 272–287. Springer, 2002.
15. 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.
16. Jean-Sébastien Coron, Marc Joye, David Naccache, and Pascal Paillier. Universal Padding Schemes for RSA. In *CRYPTO*, pages 226–241, 2002.

17. Ivan Damgård. A Design Principle for Hash Functions. In *CRYPTO*, volume 435 of *Lecture Notes in Computer Science*, pages 416–427. Springer, 1989.
18. Yevgeniy Dodis, Michael J. Freedman, Stanislaw Jarecki, and Shabsi Walfish. Versatile padding schemes for joint signature and encryption. In *ACM Conference on Computer and Communications Security*, pages 344–353, 2004.
19. Yevgeniy Dodis, Thomas Ristenpart, and Thomas Shrimpton. Salvaging Merkle-Damgård for Practical Applications. In *ePrint 2009/177 and EUROCRYPT 2009*, 2009.
20. Amos Fiat and Adi Shamir. How to Prove Yourself: Practical Solutions to Identification and Signature Problems.
21. Zheng Gong, Xuejia Lai, and Kefei Chen. A synthetic indifferentiability analysis of some block-cipher-based hash functions. In *Des. Codes Cryptography 48*, pages 293–305, 2008.
22. Jonathan J. Hoch and Adi Shamir. On the Strength of the Concatenated Hash Combiner When All the Hash Functions Are Weak. In *ICALP*, Lecture Notes in Computer Science, pages 616–630. Springer, 2008.
23. Eike Kiltz and Krzysztof Pietrzak. On the Security of Padding-Based Encryption Schemes (Or: Why we cannot prove OAEP secure in the Standard Model). In *EUROCRYPT*, pages 389–406, 2009.
24. Kazukuni Kobara and Hideki Imai. OAEP++ : A Very Simple Way to Apply OAEP to Deterministic OW-CPA Primitives. In *ePrint*, page 2002/130, 2002.
25. Yuichi Komano and Kazuo Ohta. Efficient Universal Padding Techniques for Multiplicative Trapdoor One-Way Permutation. In *CRYPTO*, pages 366–382, 2003.
26. RSA Laboratories. PKCS #1 v2.1: RSA cryptography standard. June 14, 2002.
27. Gaëtan Leurent and Phong Q. Nguyen. How Risky Is the Random-Oracle Model? In *CRYPTO*, volume 5677 of *Lecture Notes in Computer Science*, pages 445–464. Springer, 2009.
28. 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.
29. Ralph C. Merkle. One Way Hash Functions and DES. In *CRYPTO*, volume 435 of *Lecture Notes in Computer Science*, pages 428–446. Springer, 1989.
30. Yusuke Naito, Kazuki Yoneyama, Lei Wang, and Kazuo Ohta. How to Confirm Cryptosystems Security: the Original Merkle-Damgård is Still Alive! In *ASIACRYPT*, volume 5912 of *Lecture Notes in Computer Science.* Springer, 2009.
31. National Institute of Standards and Technoloty. FIPS PUB 180-3 Secure Hash Standard. In *FIPS PUB*, 2008.
32. Pascal Paillier and Jorge L. Villar. Trading one-wayness against chosen-ciphertext security in factoring-based encryption. In *ASIACRYPT*, pages 252–266, 2006.
33. Duong Hieu Phan and David Pointcheval. Chosen-Ciphertext Security without Redundancy. In *ASIACRYPT*, volume 2894 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2003.
34. Duong Hieu Phan and David Pointcheval. OAEP 3-Round: A Generic and Secure Asymmetric Encryption Padding. In *ASIACRYPT*, pages 63–77, 2004.
35. Bart Preneel, René Govaerts, and Joos Vandewalle. Hash Functions Based on Block Ciphers: A Synthetic Approach. In *CRYPTO*, volume 773 of *Lecture Notes in Computer Science*, pages 368–378. Springer, 1993.
36. Shai Halevi Ran Canetti and Jonathan Katz. A forward-secure public-key encryption scheme. In *J. Cryptology*, pages 265–294, 2007.
37. Victor Shoup. A Proposal for an ISO Standard for Public Key Encryption (version 2.1). 2001.
38. Victor Shoup. OAEP Reconsidered. In *CRYPTO*, volume 2139 of *Lecture Notes in Computer Science*, pages 239–259. Springer, 2001.
39. Martijn Stam. Blockcipher-Based Hashing Revisited. In *FSE*, volume 5665 of *Lecture Notes in Computer Science*, pages 67–83. Springer, 2009.
40. Kazuki Yoneyama, Satoshi Miyagawa, and Kazuo Ohta. Leaky Random Oracle (Extended Abstract). In *ProvSec*, volume 5324 of *Lecture Notes in Computer Science*, pages 226–240. Springer, 2008.

```
$\underline{\mathcal{F}_b(M)}$
001 If $\mathsf{F}_b(M) \neq \perp$,
002     Ret $\mathsf{F}_b(M)$;
003 $\mathsf{F}_b(M) \xleftarrow{\$} \{0,1\}^b$;
004 Ret $\mathsf{F}_b(M)$;
$\underline{EO(k,x)}$
101 If $\mathsf{EO}(k,x) \neq \perp$, ret $\mathsf{EO}(k,x)$;
102 $y \xleftarrow{\$} \{0,1\}^b$;
103 If $x = IV$, $z \leftarrow \mathcal{F}_b(k)$; $y \leftarrow z \oplus x$;
104 Else if there exists a value $M$ s.t. $\mathrm{F}_b(M) = x$,
105     $z \leftarrow \mathcal{F}_b(M||k)$; $y \leftarrow z \oplus x$;
106 $\mathsf{EO}(k,x) \leftarrow y$;
107 $\mathsf{IO}(k,y) \leftarrow x$;
108 Ret $y$;

$\underline{IO(k,y)}$
201 If $\mathsf{IO}(k,y) \neq \perp$, ret $\mathsf{D}(k,y)$;
202 $x \xleftarrow{\$} \{0,1\}^n$;
203 If $\mathsf{F}_b(k) = y \oplus IV$, $x \leftarrow IV$;
204 Else if there exists a value $M$ s.t.
    $\mathsf{F}_b(M) = z$ and $\mathsf{F}_b(M||k) = z \oplus y$,
205     $x \leftarrow z$;
206 $\mathsf{EO}(k,x) \leftarrow y$;
207 $\mathsf{IO}(k,y) \leftarrow x$;
208 Ret $x$;
```

Fig. 7. EIRO

A Indifferentiability Result for Davies-Meyer Merkle-Damård Hash Function

In this appendix, we define a new WRO called Random Oracle with Extension and Inverse Attacks (EIRO). We show that the MD hash function with Davies-Meyer compression function (denoted $\mathsf{DM\text{-}MD}$) is equal to EIRO.

A.1 Random Oracle with Extension and Inverse Attacks

The extension attack is that for the DMMD hash function $\mathsf{DM\text{-}MD}^{\mathcal{C}_{d,n}}$ $\mathsf{DM\text{-}MD}^{\mathcal{C}_{d,n}}(M||m)$ can be obtained from $\mathsf{DM\text{-}MD}^{\mathcal{C}_{d,n}}(M)$ and m without calculating $\mathsf{DM\text{-}MD}^{\mathcal{C}_{d,n}}(M||m)$. The inverse attack is that an input-output triple (k,x,y) of the ideal cipher can be obtained from $\mathsf{DM\text{-}MD}^{\mathcal{C}_{d,n}}(M)$ $(= x)$ and $\mathsf{DM\text{-}MD}^{\mathcal{C}_{d,n}}(M||k)$ $(= x \oplus y)$ without calculating $E(k,x)$ or $D(k,y)$. Therefore, we define EIRO such that $\mathcal{F}_n(M||m)$ can be obtained from $\mathcal{F}_n(M)$ and m and (k,x,y) can be obtained from $\mathcal{F}_n(M)$ and $\mathcal{F}_n(M||k)$.

The description of EIRO is shown in Fig. 7. EO is the oracle that realizes the extension attack (line 103 and line 104) and IO is the oracle that realizes the inverse attack (line 203 and line 204).

A.2 Indifferentiability Result for DMMD Hash Function in the Ideal Cipher Model

We prove that the DMMD hash function is indifferentiable from EIRO as follows.

Theorem 6. $\mathsf{DM\text{-}MD}^E \sqsubset EIRO_n$ *where for any* t_A, $t_S = t_A + \mathcal{O}(q_E + q_D)$

$$\epsilon \leq \frac{5(lq_H+q_E+q_D)^2+2(lq_H+q_E+q_D)}{2^{n+1}}$$

where A can make queries to $\mathsf{DM\text{-}MD}^{\mathcal{C}_{d,n}}/\mathcal{F}_n$, *$E/S_E$ and D/S_D at most q_H, q_E and q_D times, respectively. The maximum blocks of a* $\mathsf{DM\text{-}MD}^{\mathcal{C}_{d,n}}/\mathcal{F}_n$ *query are l blocks.*

The proof is shown in Appendix A.3.

We prove that EIRO is indifferentiable from the DMMD hash function as follows.

Theorem 7. $EIRO_n \sqsubset \mathsf{DM\text{-}MD}^{\mathcal{C}_{d,n}}$ *where for any* t_A, $t_S = t_A + \mathcal{O}(q_{EO} + q_{IO})$

$$\epsilon \leq \frac{5(lq_H+q_{EO}+q_{IO})^2+2(lq_H+q_{EO}+q_{IO})}{2^{n+1}}$$

where A can make queries to $\mathsf{DM\text{-}MD}^{\mathcal{C}_{d,n}}/\mathcal{F}_n$ at most q_H times, the maximum blocks of the query are l blocks and A can make queries to S_{EO}/EO and S_{IO}/IO at most q_{EO} and q_{IO} times, respectively. S_{EO} and S_{IO} are simulators that simulate EO and IO respectively.

The proof is shown in Appendix A.4.

A.3 Proof of Theorem 6

We define a simulator $S = (S_E, S_D)$ as follows where S_E and S_D are simulators of E and D respectively.

- $S_E(k,x)$: 001) $y \leftarrow EO(k,x)$; 002) Ret y;
- $S_D(k,y)$: 101) $x \leftarrow IO(k,y)$; 102) Ret x;

We give a proof using the game sequences Game 0, Game 1, and Game 2. In this proof, A interacts $\mathcal{O}_H$, $\mathcal{O}_E$ and $\mathcal{O}_D$.

- **Game 0**: This game is the RO scenario. Namely, $\mathcal{O}_H = \mathcal{F}_n$, $\mathcal{O}_E = S_E$ and $\mathcal{O}_D = S_D$.
- **Game 1**: In this game, we modify $\mathcal{O}_H$ where $\mathcal{O}_H = \mathsf{DM\text{-}MD}^{S_E}$. Namely $\mathsf{DM\text{-}MD}^{S_E}$ is the DMMD hash function using S_E.
- **Game 2**: This is the final game. In this game, we modify all oracles; $\mathcal{O}_H = \mathsf{DM\text{-}MD}^{E}$, $\mathcal{O}_E = E$ and $\mathcal{O}_D = D$. Namely, this game is the ideal cipher scenario.

In the following proof, an input-output triple of $\mathcal{O}_E$ and $\mathcal{O}_D$ denotes (k,x,y) where $\mathcal{O}_E(k,x) = y$ and $\mathcal{O}_D(k,y) = x$ and $w = x \oplus y$. Before starting game sequences, we define chain triples.

Definition 6 (Chain Triples). *Triples $(m_1,x_1,y_1),\ldots,(m_i,x_i,y_i)$ are chain triples if $x_1 = IV$ and $x_{j+1} = w_j$ $(j = 1,\ldots,i-1)$ hold.*

Without loss of generality, we assume that distinguisher A does not repeat a query to any of its oracles.

Game 0 → Game 1: We show that Game 0 is equal to Game 1 unless the following bad events occur.

- Event $\mathsf{E1}$: The triple (k,x,y) is such that (k,x,y) is defined by $\mathcal{O}_D$ in line 202 of IO and $x = IV$.
- Event $\mathsf{E2}$: The triple (k,x,y) is such that (k,x,y) is defined by $\mathcal{O}_D$ in line 202 of IO and there exists M such that $\mathsf{F}(M) = x$.
- Event $\mathsf{E3}$: The pair (M,z) such that $\mathcal{F}_n(M) = z$ and $z = IV$.
- Event $\mathsf{E4}$: The pairs (M,z) and (M',z') are such that $\mathcal{F}_n(M) = z$, $\mathcal{F}_n(M') = z'$ and $z = z'$.
- Event $\mathsf{E5}$: The pair (M,z) is such that $\mathcal{F}_n(M) = z$ and there exists a triple (k,x,y) such that $z = x$ and the triple is defined in line 102 of EO or 202 of IO.

In order to prove that Game 0 is equal to Game 1 unless the following bad events occur, we show the following three points.

1. In Game 0, unless a bad event occurs, the answers given by $\mathcal{O}_E$ and $\mathcal{O}_D$ are consistent with those given by $\mathcal{O}_H$.

2. In Game 1, unless a bad event occurs, the answers given by $\mathcal{O}_E$ and $\mathcal{O}_D$ are consistent with those given by $\mathcal{O}_H$.
3. Unless a bad event occurs, for any M $\mathcal{O}_H(M) = \mathcal{F}_n(M)$ in Game 0 and Game 1.

If the above three points hold, $|Pr[G1] - Pr[G0]| \leq Pr[\mathsf{E1} \vee \mathsf{E2} \vee \mathsf{E3} \vee \mathsf{E4} \vee \mathsf{E5}] \leq Pr[\mathsf{E1}] + Pr[\mathsf{E2}] + Pr[\mathsf{E3}] + Pr[\mathsf{E4}] + Pr[\mathsf{E5}]$. So we also show that $Pr[\mathsf{E1}], Pr[\mathsf{E2}], Pr[\mathsf{E3}], Pr[\mathsf{E4}]$ and $Pr[\mathsf{E5}]$ are negligible.

Before starting the proof of the above points, we give a useful lemma.

Lemma 3. *For any chain triples $(k_1, x_1, y_1), \ldots, (k_i, x_i, y_i)$ defined by $\mathcal{O}_E$ or $\mathcal{O}_D$, unless a bad event occurs, $w_i = \mathcal{F}_n(k_1 || \cdots || k_i)$.*

Proof. To the contrary, assume that there exist chain triples $(k_1, x_1, y_1), \ldots, (k_i, x_i, y_i)$ defined by $\mathcal{O}_E$ or $\mathcal{O}_D$ such that $w_i \neq \mathcal{F}_n(k_1 || \cdots || k_i)$.

We consider two cases: (Case 1) $\forall j \in \{1, \ldots, i\} : w_j \neq \mathcal{F}_n(k_1 || \cdots || k_j)$. (Case 2) $\exists j \in \{1, \ldots, i-1\}$ such that $w_j = \mathcal{F}_n(k_1 || \cdots || k_j)$ (Note that since $w_i \neq \mathcal{F}_n(k_1 || \cdots || k_i)$, $j \neq i$).

We consider Case 1. From the condition of this case, $w_1 \neq \mathcal{F}_n(k_1)$ holds. (k_1, x_1, y_1) is defined by EO or IO. Since $x_1 = IV$, if (k_1, x_1, y_1) is defined by EO, (k_1, x_1, y_1) is defined in line 103 of EO. Therefore, in this case $w_1 = \mathcal{F}_n(k_1)$. This contradicts Case 1. If (k_1, x_1, y_1) is defined by IO, since $x_1 = IV$ and $w_1 \neq \mathcal{F}_n(k_1)$, this triple is defined in line 202 of IO. Therefore, event $\mathsf{E1}$ occurs.

We consider Case 2. We assume that j is the maximum number in $\{1, \ldots, i-1\}$ such that $w_j = \mathcal{F}_n(k_1 || \cdots || k_j)$ holds. We divide Case 2 into two cases: (Case 2-1) $(k_{j+1}, x_{j+1}, y_{j+1})$ is defined by $\mathcal{F}_n$. (Case 2-2) $(k_{j+1}, x_{j+1}, y_{j+1})$ is not defined by $\mathcal{F}_n$.

We consider Case 2-1. In this case, $\exists M$ such that $w_{j+1} = \mathcal{F}_n(M || k_{j+1})$. From the condition of j, $M \neq k_1 || \cdots || k_j$ holds. We divide Case 2-1 into two cases: (Case 2-1-1) $M = \perp$. (Case 2-1-2) $M \neq \perp$.

In Case 2-1-1, $w_{j+1} = \mathcal{F}_n(k_{j+1})$ holds. From the definition of EIRO, $(k_{j+1}, x_{j+1}, y_{j+1})$ is defined by $\mathcal{F}_n$ in line 103 of EO, 105 of EO, 203 of IO or 204 of IO. Since $M = \perp$, the line is 103 of EO or 203 of IO. From the condition of executing line 103 of EO or line 203 of IO, $x_{j+1} = IV$ holds. Since $x_{j+1} = w_j = \mathcal{F}_n(k_1 || \cdots || k_j)$ and $x_{j+1} = IV$ hold, event $\mathsf{E3}$ occurs.

In Case 2-1-2, $M \neq \perp$ holds. From the definition of EIRO, $(k_{j+1}, x_{j+1}, y_{j+1})$ is defined by $\mathcal{F}_n$ in line 103 of EO, 104 of EO, 203 of IO or 204 of IO. Since $M \neq \perp$ holds, the line is 104 of EO or 204 of IO. From the condition of executing line 104 of EO or line 204 of IO, $x_{j+1} = \mathrm{RO}(M)$ holds. Since $x_{j+1} = w_j = \mathcal{F}_n(k_1 || \cdots || k_j)$ and $x_{j+1} = \mathcal{F}_n(M)$ holds, event $\mathsf{E4}$ occurs.

We consider Case 2-2. Since $(k_{j+1}, x_{j+1}, y_{j+1})$ is not defined by $\mathcal{F}_n$, the triple is defined in line 102 of EO or 202 of IO. We consider the case that $(k_{j+1}, x_{j+1}, y_{j+1})$ is defined in line 102 of EO. In this case, since $x_{j+1} = w_j = \mathcal{F}_n(k_1 || \cdots || k_j)$ holds, when (k_j, x_j, y_j) is defined, $(k_{j+1}, x_{j+1}, y_{j+1})$ is already defined (If (m_j, x_j, y_j) is defined before defining $(kj + 1, x_{j+1}, y_{j+1})$, $w_{j+1} = \mathcal{F}_n(k_1 || \cdots || k_{j+1})$ holds from line 104 of EO). Therefore in this case event $\mathsf{E5}$. Finally, we consider the case that $(k_{j+1}, x_{j+1}, y_{j+1})$ is defined in line 202 of IO. This case occurs in event $\mathsf{E2}$ or $\mathsf{E5}$.

The proof is completed. □

By using the lemma, we prove the three points.

First we prove the first point. From Lemma 3, for any chain triples $(k_1, x_1, y_1), \ldots, (k_i, x_i, y_i)$, unless a bad event occurs, $w_i = \mathcal{F}_n(k_1 || \cdots || k_i)$. Since $\mathcal{O}_H = \mathcal{F}_n$, the answers given by $\mathcal{O}_E$ and $\mathcal{O}_D$ are consistent with those given by $\mathcal{O}_H$.

We prove the second point. Since $\mathcal{O}_H$ uses $\mathcal{O}_E$ ($\mathcal{O}_H = \mathsf{DM\text{-}MD}^{\mathcal{O}_E}$), the answers given by $\mathcal{O}_E$ and $\mathcal{O}_D$ are consistent with those given by $\mathcal{O}_H$.

We prove the third point. From Lemma 3, unless a bad event occurs, in Game 1 for any M $\mathcal{O}_H(M) = \mathcal{F}_n(M)$. And in Game 0 $\mathcal{O}_H = \mathcal{F}_n$.

Thus Game 1 is equal to Game 0 unless a bad event occurs.

Next we bound the probabilities $Pr[\mathsf{E1}], Pr[\mathsf{E2}], Pr[\mathsf{E3}], Pr[\mathsf{E4}]$ and $Pr[\mathsf{E5}]$.

- $Pr[\mathsf{E1}]$: An output of S_D is chosen uniformly from $\{0,1\}^n$. Since the maximum number of times that $\mathcal{O}_D$ is called is q_D, $Pr[\mathsf{E1}] \leq \frac{q_D}{2^n}$.
- $Pr[\mathsf{E2}]$: Since an output of $\mathcal{F}_n$ is chosen uniformly from $\{0,1\}^n$ and the maximum number of times that $\mathcal{F}_n$ is called is $lq_H + q_E + q_D$, $Pr[\mathsf{E2}] \leq \frac{(lq_H+q_E+q_D)q_D}{2^n}$.
- $Pr[\mathsf{E3}]$: Since an output of $\mathcal{F}_n$ is chosen uniformly from $\{0,1\}^n$ and the maximum number of times that $\mathcal{F}_n$ is called is $lq_H + q_E + q_D$, $Pr[\mathsf{E3}] \leq \frac{lq_H+q_E+q_D}{2^n}$.
- $Pr[\mathsf{E4}]$: Since an output of $\mathcal{F}_n$ is chosen uniformly from $\{0,1\}^n$ and the maximum number of times that $\mathcal{F}_n$ is called is $lq_H + q_E + q_D$, $Pr[\mathsf{E4}] \leq \frac{(lq_H+q_E+q_D)^n}{2^n}$.
- $Pr[\mathsf{E5}]$: Since (k', x', y') is defined in line 102 or 202, the triple is defined independently from $\mathcal{F}_n$. Since an output of $\mathcal{F}_n$ is chosen uniformly from $\{0,1\}^n$ and the maximum number of times that $\mathcal{F}_n$ is called is $lq_H + q_E + q_D$, $Pr[\mathsf{E5}] \leq \frac{(lq_H+q_E+q_D)(q_E+q_D)}{2^n}$

Therefore, $|Pr[G1] - Pr[G0] \leq \frac{2(lq_H+q_E+q_D)^2+lq_H+q_E+q_D}{2^n}$.

Game 1 → Game 2: Since outputs of S_E and S_D are chosen uniformly from $\{0,1\}^n$, $S_E = E$ and $S_D = D$ unless a collision occurs. Thus we have via a straightforward birthday analysis that $|Pr[G2] - Pr[G1]| \leq \frac{(lq_H+q_E+q_D)^2}{2^{n+1}}$.

The proof of the theorem is completed. □

A.4 Proof of Theorem 7

We define simulator $\mathsf{S} = (S_{EO}, S_{IO})$ as follows.

Simulator S:

$S_{EO}(m,x)$: $y \leftarrow E(m,x)$ and S returns y.

$S_{IO}(m,y)$, $x \leftarrow E^{-1}(m,y)$ and S returns x.

The running time of S is at most $O(q_E)$ time.

This proof utilizes the proof of Theorem 6. The proof involves a hybrid argument starting in the EIRO scenario, and ending in the ideal cipher scenario through a sequence of mutually indistinguishable hybrid games.

Game 0. This game is the same as the EIRO scenario. Let $G0$ be the event that A outputs 1 in this game. $Pr[G0] = Pr[D^{\mathsf{EIRO}} \Rightarrow 1]$ holds.

Game 1. In this game, A interacts with $(\mathsf{DM\text{-}MD}^E, \mathcal{C}_{d,n})$. In the proof of Theorem 6, $\mathsf{S}_E(m,x)$ returns the output of $EO(m,x)$, and $\mathsf{S}_D(m,y)$ returns the output of $IO(m,y)$. Therefore, the view of A in Game 0 is identical with that of A in Game 0 of the proof of Theorem 6. Game 1 is identical with Game 2 in the proof of Theorem 6. Let $G1$ be the event that A outputs 1 in this game. From the proof of Theorem 6, $|Pr[G1] - Pr[G0]| \leq \frac{5(lq_H+q_{EO}+q_{IO})^2+2(lq_H+q_{EO}+q_{IO})}{2^{n+1}}$.

Game 2. This is the final game. In this game, A interacts with $(\mathsf{DM\text{-}MD}^E, \mathsf{S})$. Let $G2$ be the event that A outputs 1 in this game. Since for a query S_E simply returns the output of E and for a query S_D simply returns the output of D, $Pr[\mathsf{G2}] = Pr[\mathsf{G1}]$.

Now we can complete the proof of Theorem 6 by combining Games 0 to 2, and observing that Game 1 is the same as EIRO scenario while Game 3 is same as $\mathsf{DM\text{-}MD}^E$ scenario. Hence we can deduce that $\epsilon \leq \frac{5(lq_H+q_{EO}+q_{IO})^2+2(lq_H+q_{EO}+q_{IO})}{2^{n+1}}$. □