Authenticated Key Exchange and Key Encapsulation Without Random Oracles

Tatsuaki Okamoto

NTT, 3-9-11 Midori-cho, Musashino-shi, Tokyo, 180-8585 Japan
December 19, 2007

Abstract. This paper[1] presents a new paradigm to realize cryptographic primitives such as authenticated key exchange and key encapsulation without random oracles under three assumptions: the decisional Diffie-Hellman (DDH) assumption, target collision resistant (TCR) hash functions and a class of pseudo-random functions (PRFs), πPRFs, PRFs with pairwise-independent random sources. We propose a (PKI-based) two-pass authenticated key exchange (AKE) protocol that is comparably as efficient as the existing most efficient protocols like MQV and that is secure without random oracles (under these assumptions). Our protocol is shown to be secure in the (currently) strongest security definition, the extended Canetti-Krawczyk (eCK) security definition introduced by LaMacchia, Lauter and Mityagin. We also show that a variant of the Kurosawa-Desmedt key encapsulation mechanism (KEM) using a πPRF is CCA-secure. This scheme is secure in a stronger security notion, the chosen public-key and ciphertext attack (CPCA) security, with using generalized TCR (GTCR) hash functions. The proposed schemes in this paper are redundancy-free (or validity-check-free) and the implication is that combining them with redundancy-free symmetric encryption (DEM) will yield redundancy-free (e.g., MAC-free) CCA-secure hybrid encryption.

1 Introduction

The most common paradigm to design practical public-key cryptosystems secure in the standard model is to combine a trapdoor function (e.g., Diffie-Hellman or RSA function) and target collision resistance (TCR) hash functions, where the security is proven under a trapdoor function assumption (e.g., DDH or SRSA assumption) and the TCR hash function assumption [1, 3, 9].

This paper introduces a paradigm to design practical public-key cryptosystems, where a class of *pseudo-random function* (PRF), πPRFs, PRFs with pairwise-independent random sources, is employed in addition to a trapdoor function (DH) and target collision resistant (TCR) hash function.

Authenticated key exchange (AKE) protocols have been extensively studied to enhance the security of the Diffie-Hellman (DH) key exchange protocol, which was proposed in 1976, because the DH protocol is not secure against the man-in-the-middle attack [2, 8, 10–13, 17].

[1] This is a revised version of the extended abstract appeared in the proceedings of Asiacrypt 2007 [15].

This paper presents a (PKI-based) two-pass AKE protocol that offers the following properties:

1. its efficiency is comparable to those of MQV [11], HMQV [8] and CMQV [17] (our scheme's message size for one party is that of MQV plus the size of three group elements, and the computational complexity for a session of our scheme is around 4.3 group exponentiations, while that of MQV is around 2.2 group exponentiations),
2. the model for its security proof is not the random oracle model,
3. its underlying security definition is (currently) the strongest one, the extended Canetti-Krawczyk (eCK) security definition introduced by LaMacchia, Lauter and Mityagin [10],
4. its security proof reduction efficiency is better than those of previous protocols in the random oracle model.

This paper also proposes a *CCA-secure* key encapsulation mechanism (KEM) under these assumptions, which is a variant of the Kurosawa-Desmedt KEM [9]. This scheme is also secure in a stronger security notion, the *chosen public-key and ciphertext attack (CPCA)* security, in which an adversary, given a target public key pk^* and ciphertext c^*, is allowed to query a pair of public key pk and ciphertext c to the decryption oracle, which answers the adversary with the decrypted result of c by the secret key of pk.

The proposed schemes in this paper are redundancy-free (or validity-check-free) and implies redundancy-free (e.g., MAC-free) CCA-secure hybrid encryption by combining with redundancy-free CCA-secure symmetric encryption (DEM).

2 Preliminaries

2.1 Notations

$\mathbb{N}$ is the set of natural numbers and $\overline{\mathbb{R}}$ is the set of real numbers. $\perp$ denotes a null string.

A function $f : \mathbb{N} \rightarrow \overline{\mathbb{R}}$ is *negligible* in k, if for every constant $c > 0$, there exists integer n such that $f(k) < k^{-c}$ for all $k > n$. Hereafter, we often use $f(k) < \epsilon(k)$ to mean that f is negligible in k.

When A is a probabilistic machine or algorithm, $A(x)$ denotes the random variable of A's output on input x. Then, $y \stackrel{\mathsf{R}}{\leftarrow} A(x)$ denotes that y is randomly selected from $A(x)$ according to its distribution. When a is a value, $A(x) \rightarrow a$ denotes the event that A outputs a on input x. When A is a set, $y \stackrel{\mathsf{U}}{\leftarrow} A$ denotes that y is uniformly selected from A. When A is a value, $y \leftarrow A$ denotes that y is set as A.

In this paper, we consider that the underlying machines are uniform Turing machines. But it is easy to extend our results to non-uniform Turing machines.

2.2 The DDH Assumption

Let k be a security parameter and $\mathbb{G}$ be a group with security parameter k, where the order of $\mathbb{G}$ is prime p and $|p| = k$. Let $\{\mathbb{G}\}_k$ be the set of group $\mathbb{G}$ with security parameter k.

For all $k \in \mathbb{N}$ we define the sets $\mathbb{D}$ and $\mathbb{R}$ as follows:

$$\mathbb{D}(k) \leftarrow \{(\mathbb{G}, g_1, g_2, g_1^x, g_2^x) \mid \mathbb{G} \stackrel{\mathsf{U}}{\leftarrow} \{\mathbb{G}\}_k, (g_1, g_2) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{G}^2, x \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p\}$$
$$\mathbb{R}(k) \leftarrow \{(\mathbb{G}, g_1, g_2, y_1, y_2) \mid \mathbb{G} \stackrel{\mathsf{U}}{\leftarrow} \{\mathbb{G}\}_k, (g_1, g_2, y_1, y_2) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{G}^4\}.$$

Let $\mathcal{A}$ be a probabilistic polynomial-time machine. For all $k \in \mathbb{N}$, we define the DDH advantage of $\mathcal{A}$ as

$$\mathsf{Adv}\mathrm{DDH}_{\mathcal{A}}(k) \leftarrow |\ \Pr[\mathcal{A}(1^k, \rho) \to 1 \mid \rho \stackrel{\mathsf{U}}{\leftarrow} \mathbb{D}(k)] - \Pr[\mathcal{A}(1^k, \rho) \to 1 \mid \rho \stackrel{\mathsf{U}}{\leftarrow} \mathbb{R}(k)]\ |.$$

The DDH assumption for $\{\mathbb{G}\}_{k \in \mathbb{N}}$ is: For any probabilistic polynomial-time adversary $\mathcal{A}$, $\mathsf{Adv}\mathrm{DDH}_{\mathcal{A}}(k)$ is negligible in k.

2.3 Pseudo-Random Function (PRF)

The concept of a pseudo-random function (PRF) is defined in [5] by Goldwasser, Goldreich and Micali.

Let $k \in \mathbb{N}$ be a security parameter. A pseudo-random function (PRF) family F associated with $\{\mathsf{Seed}_k\}_{k \in \mathbb{N}}$, $\{\mathsf{Dom}_k\}_{k \in \mathbb{N}}$ and $\{\mathsf{Rng}_k\}_{k \in \mathbb{N}}$ specifies two items:

– A family of random seeds $\{\mathsf{Seed}_k\}_{k \in \mathbb{N}}$.
– A family of pseudo-random functions indexed by k, $\Sigma \stackrel{\mathsf{R}}{\leftarrow} \mathsf{Seed}_k$, $\sigma \stackrel{\mathsf{U}}{\leftarrow} \Sigma$, $\mathcal{D} \stackrel{\mathsf{R}}{\leftarrow} \mathsf{Dom}_k$, and $\mathcal{R} \stackrel{\mathsf{R}}{\leftarrow} \mathsf{Rng}_k$, where each such function $\mathsf{F}_\sigma^{k,\Sigma,\mathcal{D},\mathcal{R}}$ maps an element of $\mathcal{D}$ to an element of $\mathcal{R}$. There must exist a deterministic polynomial-time algorithm that on input 1^k, σ and ρ, outputs $\mathsf{F}_\sigma^{k,\Sigma,\mathcal{D},\mathcal{R}}(\rho)$.

Let $\mathcal{A}^O$ be a probabilistic polynomial-time machine with oracle access to O. For all k, we define

$$\mathsf{Adv}\mathrm{PRF}_{\mathsf{F},\mathcal{A}}(k) \leftarrow |\Pr[\mathcal{A}^F(1^k, \mathcal{D}, \mathcal{R}) \to 1] - \Pr[\mathcal{A}^{RF}(1^k, \mathcal{D}, \mathcal{R}) \to 1]|,$$

where $\Sigma \stackrel{\mathsf{R}}{\leftarrow} \mathsf{Seed}_k$, $\sigma \stackrel{\mathsf{U}}{\leftarrow} \Sigma$, $\mathcal{D} \stackrel{\mathsf{R}}{\leftarrow} \mathsf{Dom}_k, \mathcal{R} \stackrel{\mathsf{R}}{\leftarrow} \mathsf{Rng}_k$, $F \leftarrow \mathsf{F}_\sigma^{k,\Sigma,\mathcal{D},\mathcal{R}}$, and $RF : \mathcal{D} \to \mathcal{R}$ is a truly random function ($\forall \rho \in \mathcal{D} \quad RF(\rho) \stackrel{\mathsf{U}}{\leftarrow} \mathcal{R}$).

F is a pseudo-random function (PRF) family if for any probabilistic polynomial-time adversary $\mathcal{A}$, $\mathsf{Adv}\mathrm{PRF}_{\mathsf{F},\mathcal{A}}(k)$ is negligible in k.

2.4 Pseudo-Random Function with Pairwise-Independent Random Sources (πPRF)

Here, we introduce a specific class of PRFs, πPRFs.

Let $k \in \mathbb{N}$ be a security parameter and F be a PRF family associated with $\{\mathsf{Seed}_k\}_{k \in \mathbb{N}}$, $\{\mathsf{Dom}_k\}_{k \in \mathbb{N}}$ and $\{\mathsf{Rng}_k\}_{k \in \mathbb{N}}$.

We then define a πPRF family for F.

Let $\Sigma \stackrel{\mathsf{R}}{\leftarrow} \mathsf{Seed}_k$, $\mathcal{D} \stackrel{\mathsf{R}}{\leftarrow} \mathsf{Dom}_k, \mathcal{R} \stackrel{\mathsf{R}}{\leftarrow} \mathsf{Rng}_k$, and $RF : \mathcal{D} \to \mathcal{R}$ is a truly random function ($\forall \rho \in \mathcal{D} \quad RF(\rho) \stackrel{\mathsf{U}}{\leftarrow} \mathcal{R}$).

Let X_Σ be a set of random variables (distributions) over Σ, and I_Σ be a set of indices regarding Σ such that there exists a deterministic polynomial-time algorithm, $f_\Sigma : I_\Sigma \rightarrow X_\Sigma$, that on input $i \in I_\Sigma$, outputs $\sigma_i \in X_\Sigma$.

Let $(\sigma_{i_0}, \sigma_{i_1}, \ldots, \sigma_{i_{t(k)}})$ be random variables indexed by (I_Σ, f_Σ), where $i_j \in I_\Sigma$ ($j = 0, 1, \ldots, t(k)$) and $t(k)$ is a polynomial of k. Let σ_{i_0} be pairwisely independent from other variables, $\sigma_{i_1}, \ldots, \sigma_{i_{t(k)}}$, and each variable be uniformly distributed over Σ. That is, for any pair of $(\sigma_{i_0}, \sigma_{i_j})$ ($j = 1, \ldots, t(k)$), for any $(x, y) \in \Sigma^2$, $\Pr[\sigma_{i_0} \rightarrow x \wedge \sigma_{i_j} \rightarrow y] = \Pr[\sigma_{i_0} \rightarrow x] \cdot \Pr[\sigma_{i_j} \rightarrow y] = 1/|\Sigma|^2$.

Let $\mathcal{A}^{F, I_\Sigma}$ be a probabilistic polynomial-time machine $\mathcal{A}$ that queries $q_j \in \mathcal{D}$ along with $i_j \in I_\Sigma$ to oracle (F, I_Σ) and is replied with $\mathsf{F}^{k,\Sigma,\mathcal{D},\mathcal{R}}_{\overline{\sigma}_j}(q_j)$ for each $j = 0, 1, \ldots, t(k)$, where $(\overline{\sigma}_0, \ldots, \overline{\sigma}_{t(k)}) \stackrel{\mathsf{R}}{\leftarrow} (\sigma_{i_0}, \ldots, \sigma_{i_{t(k)}})$ in oracle (F, I_Σ).

Let $\mathcal{A}^{RF, I_\Sigma}$ be the same as $\mathcal{A}^{F, I_\Sigma}$ except $\mathsf{F}^{k,\Sigma,\mathcal{D},\mathcal{R}}_{\overline{\sigma}_0}(q_0)$ is replaced by $RF(q_0)$.

For all k, we define

$$\mathsf{Adv}\pi\mathrm{PRF}_{\mathsf{F}, I_\Sigma, \mathcal{A}}(k) \leftarrow |\Pr[\mathcal{A}^{F, I_\Sigma}(1^k, \mathcal{D}, \mathcal{R}) \rightarrow 1] - \Pr[\mathcal{A}^{RF, I_\Sigma}(1^k, \mathcal{D}, \mathcal{R}) \rightarrow 1]|.$$

F is a πPRF family with index $\{(I_\Sigma, f_\Sigma)\}_{\Sigma \in \mathsf{Seed}_k, k \in \mathbb{N}}$ if for any probabilistic polynomial-time adversary $\mathcal{A}$, $\mathsf{Adv}\pi\mathrm{PRF}_{\mathsf{F}, I_\Sigma, \mathcal{A}}(k)$ is negligible in k.

Remark: Here, we introduce an example of index (I_Σ, f_Σ) for pairwisely independent random variables, which is used in the proposed schemes.

Let k be a security parameter and $\mathbb{G}$ be a group with security parameter k, where the order of $\mathbb{G}$ is prime p and $|p| = k$. Let $\Sigma \leftarrow \mathbb{G}$. Then $(I_\mathbb{G}, f_\mathbb{G})$ is specified by

$$I_\mathbb{G} \leftarrow \{(V, W, d) \mid (V, W, d) \in \mathbb{G}^2 \times \mathbb{Z}_p\},$$
$$X_\mathbb{G} \leftarrow \{\sigma_{(V,W,d)} \mid \sigma_{(V,W,d)} \leftarrow V^{r_1 + dr_2} W \wedge (V, W, d) \in \mathbb{G}^2 \times \mathbb{Z}_p \wedge (r_1, r_2) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2\},$$
$$f_\mathbb{G} : I_\mathbb{G} \rightarrow X_\mathbb{G} \quad \text{and} \quad f_\mathbb{G} : (V, W, d) \mapsto \sigma_{(V,W,d)}.$$

If $d \neq d'$, $V \neq 1$ and $V' \neq 1$, then two random variables, $\sigma_{(V,W,d)} \in X_\mathbb{G}$ and $\sigma_{(V',W',d')} \in X_\mathbb{G}$, are pairwisely independent, and each one is uniformly distributed over $\mathbb{G}$, whereas three random variables, $\sigma_{(V,W,d)} \in X_\mathbb{G}$, $\sigma_{(V',W',d')} \in X_\mathbb{G}$ and $\sigma_{(V'',W'',d'')} \in X_\mathbb{G}$, are not independent.

In the experiment of defining $\mathsf{Adv}\pi\mathrm{PRF}_{\mathsf{F}, I_\mathbb{G}, \mathcal{A}}(k)$, $\mathcal{A}^{F, I_\mathbb{G}}$ queries $q_j \in \mathcal{D}$ along with $(V_j, W_j, d_j) \in I_\mathbb{G}$ to oracle $(F, I_\mathbb{G})$ and is replied with $\mathsf{F}^{k,\Sigma,\mathcal{D},\mathcal{R}}_{\overline{\sigma}_j}(q_j)$ for each $j = 0, 1, \ldots, t(k)$, where $(\overline{\sigma}_0, \ldots, \overline{\sigma}_{t(k)}) \stackrel{\mathsf{R}}{\leftarrow} (\sigma_{(V_0, W_0, d_0)}, \ldots, \sigma_{(V_{t(k)}, W_{t(k)}, d_{t(k)})})$ and the random selection of $(\overline{\sigma}_0, \ldots, \overline{\sigma}_{t(k)})$ is due to the selection of $(r_1, r_2) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$ in oracle $(F, I_\mathbb{G})$.

Hereafter, this index, $(I_\mathbb{G}, f_\mathbb{G})$, is shortly expressed by $I_\mathbb{G} \leftarrow \{(V, W, d) \mid (V, W, d) \in \mathbb{G}^2 \times \mathbb{Z}_p\}$ and $f_\mathbb{G} : (V, W, d) \mapsto V^{r_1 + dr_2} W$ with $(r_1, r_2) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$.

2.5 Target Collision Resistant (TCR) Hash Function

Let $k \in \mathbb{N}$ be a security parameter. A target collision resistant (TCR) hash function family H associated with $\{\mathsf{Dom}_k\}_{k \in \mathbb{N}}$ and $\{\mathsf{Rng}_k\}_{k \in \mathbb{N}}$ specifies two items:

– A family of key spaces indexed by k. Each such key space is a probability space on bit strings denoted by KH_k. There must exist a probabilistic polynomial-time algorithm whose output distribution on input 1^k is equal to KH_k.
– A family of hash functions indexed by k, $h \stackrel{\mathsf{R}}{\leftarrow} \mathsf{KH}_k$, $\mathcal{D} \stackrel{\mathsf{R}}{\leftarrow} \mathsf{Dom}_k$, and $\mathcal{R} \stackrel{\mathsf{R}}{\leftarrow} \mathsf{Rng}_k$, where each such function $\mathsf{H}_h^{k,\mathcal{D},\mathcal{R}}$ maps an element of $\mathcal{D}$ to an element of $\mathcal{R}$. There must exist a deterministic polynomial-time algorithm that on input 1^k, h and ρ, outputs $\mathsf{H}_h^{k,\mathcal{D},\mathcal{R}}(\rho)$.

Let $\mathcal{A}$ be a probabilistic polynomial-time machine. For all k, we define

$$\mathsf{Adv}\mathrm{TCR}_{\mathsf{H},\mathcal{A}}(k) \leftarrow$$
$$\Pr[\rho \in \mathcal{D} \wedge \rho \neq \rho^* \wedge \mathsf{H}_h^{k,\mathcal{D},\mathcal{R}}(\rho) = \mathsf{H}_h^{k,\mathcal{D},\mathcal{R}}(\rho^*) \ \mid\ \rho \stackrel{\mathsf{R}}{\leftarrow} \mathcal{A}(1^k, \rho^*, h, \mathcal{D}, \mathcal{R})],$$

where $\mathcal{D} \stackrel{\mathsf{R}}{\leftarrow} \mathsf{Dom}_k, \mathcal{R} \stackrel{\mathsf{R}}{\leftarrow} \mathsf{Rng}_k, \rho^* \stackrel{\mathsf{U}}{\leftarrow} \mathcal{D}$ and $h \stackrel{\mathsf{R}}{\leftarrow} \mathsf{KH}_k$. H is a target collision resistance (TCR) hash function family if for any probabilistic polynomial-time adversary $\mathcal{A}$, $\mathsf{Adv}\mathrm{TCR}_{\mathsf{H},\mathcal{A}}(k)$ is negligible in k.

2.6 PKI-Based Authenticated Key Exchange (AKE) and the Extended Canetti-Krawczyk (eCK) Security Definition

This section outlines the extended Canetti-Krawczyk (eCK) security definition for two pass PKI-based authenticated key exchange (AKE) protocols that was introduced by LaMacchia, Lauter and Mityagin [10], and follows the description in [17].

In the eCK definition, we suppose there are n parties which are modeled as probabilistic polynomial-time Turing machines. We assume that some agreement on the common parameters in the AKE protocol has been made among the parties before starting the protocol. The mechanism by which these parameters are selected is out of scope of the AKE protocol and the (eCK) security model.

Each party has a static public-private key pair together with a certificate that binds the public key to that party. $\hat{A}$ ($\hat{B}$) denotes the static public key A (B) of party $\mathcal{A}$ ($\mathcal{B}$) together with a certificate. We do not assume that the certifying authority (CA) requires parties to prove possession of their static private keys, but we require that the CA verifies that the static public key of a party belongs to the domain of public keys.

Here, two parties exchange static public keys A, B and ephemeral public keys X, Y; the session key is obtained by combining A, B, X, Y and possibly session identities. A party $\mathcal{A}$ can be activated to execute an instance of the protocol called a *session*. Activation is made via an incoming message that has one of the following forms: $(\hat{A}, \hat{B})$ or $(\hat{B}, \hat{A}, X)$. If $\mathcal{A}$ was activated with $(\hat{A}, \hat{B})$, then $\mathcal{A}$ is called the session initiator, otherwise the session responder. Session initiator $\mathcal{A}$ creates ephemeral public-private key pair, (X, x) and sends $(\hat{B}, \hat{A}, X)$ to session responder $\mathcal{B}$. $\mathcal{B}$ then creates ephemeral public-private key pair, (Y, y) and sends $(\hat{A}, \hat{B}, X, Y)$ to $\mathcal{A}$.

The session of initiator $\mathcal{A}$ with responder $\mathcal{B}$ is identified via session identifier $(\hat{A}, \hat{B}, X, Y)$, where $\mathcal{A}$ is said the owner of the session, and $\mathcal{B}$ the peer of the session. The session of responder $\mathcal{B}$ with initiator $\mathcal{A}$ is identified as $(\hat{B}, \hat{A}, Y, X)$, where $\mathcal{B}$ is the owner, and $\mathcal{A}$ is the peer. Session $(\hat{B}, \hat{A}, Y, X)$ is said a matching session of $(\hat{A}, \hat{B}, X, Y)$. We say that a session is completed if its owner computes a session key.

The adversary $\mathcal{M}$ is modeled as a probabilistic polynomial-time Turing machine and controls all communications. Parties submit outgoing messages to the adversary, who makes decisions about their delivery. The adversary presents parties with incoming messages via $\mathsf{Send}(message)$, thereby controlling the activation of sessions. In order to capture possible leakage of private information, adversary $\mathcal{M}$ is allowed the following queries:

– $\mathsf{EphemeralKeyReveal}(\mathsf{sid})$ The adversary obtains the ephemeral private key associated with session sid.
– $\mathsf{SessionKeyReveal}(\mathsf{sid})$ The adversary obtains the session key for session sid, provided that the session holds a session key.
– $\mathsf{StaticKeyReveal}(\mathsf{pid})$ The adversary learns the static private key of party pid.
– $\mathsf{EstablishParty}(\mathsf{pid})$ This query allows the adversary to register a static public key on behalf of a party. In this way the adversary totally controls that party.

If a party pid is established by $\mathsf{EstablishParty}(\mathsf{pid})$ query issued by adversary $\mathcal{M}$, then we call the party *dishonest*. If a party is not dishonest, we call the party *honest*.

The aim of adversary $\mathcal{M}$ is to distinguish a session key from a random key. Formally, the adversary is allowed to make a special query $\mathsf{Test}(\mathsf{sid}^*)$, where sid^* is called the *target session*. The adversary is then given with equal probability either the session key, K^*, held by sid^* or a random key, $R^* \overset{\mathsf{U}}{\leftarrow} \{0,1\}^{|K^*|}$. The adversary wins the game if he guesses correctly whether the key is random or not. To define the game, we need the notion of *fresh session* as follows:

Definition 1. *(fresh session) Let* sid *be the session identifier of a completed session, owned by an honest party* $\mathcal{A}$ *with peer* $\mathcal{B}$*, who is also honest. Let* $\overline{\mathsf{sid}}$ *be the session identifier of the matching session of* sid*, if it exists. Define session* sid *to be "fresh" if none of the following conditions hold:*

– $\mathcal{M}$ *issues a* $\mathsf{SessionKeyReveal}(\mathsf{sid})$ *query or a* $\mathsf{SessionKeyReveal}(\overline{\mathsf{sid}})$ *query (if* $\overline{\mathsf{sid}}$ *exists),*
– $\overline{\mathsf{sid}}$ *exists and* $\mathcal{M}$ *makes either of the following queries:*
both $\mathsf{StaticKeyReveal}(\mathcal{A})$ *and* $\mathsf{EphemeralKeyReveal}(\mathsf{sid})$*, or*
both $\mathsf{StaticKeyReveal}(\mathcal{B})$ *and* $\mathsf{EphemeralKeyReveal}(\overline{\mathsf{sid}})$*,*
– $\overline{\mathsf{sid}}$ *does not exist and* $\mathcal{M}$ *makes either of the following queries:*
both $\mathsf{StaticKeyReveal}(\mathcal{A})$ *and* $\mathsf{EphemeralKeyReveal}(\mathsf{sid})$*, or*
$\mathsf{StaticKeyReveal}(\mathcal{B})$*.*

We are now ready to present the eCK security notion.

Definition 2. *(eCK security) Let* K^* *be a session key of the target session* sid^* *that should be "fresh",* $R^* \overset{\mathsf{U}}{\leftarrow} \{0,1\}^{|K^*|}$*, and* $b^* \overset{\mathsf{U}}{\leftarrow} \{0,1\}$*. As a reply to* $\mathsf{Test}(\mathsf{sid}^*)$ *query by* $\mathcal{M}$*,* K^* *is given to* $\mathcal{M}$ *if* $b^* = 0$*;* R^* *is given otherwise. Finally* $\mathcal{M}$ *outputs* $b \in \{0,1\}$*. We define*

$$\mathsf{Adv}\mathrm{AKE}_{\mathcal{M}}(k) \leftarrow |\Pr[b = b^*] - 1/2|.$$

A key exchange protocol is secure if the following conditions hold:

– *If two honest parties complete matching sessions, then they both compute the same session key (or both output indication of protocol failure).*
– *For any probabilistic polynomial-time adversary $\mathcal{M}$,* $\mathsf{Adv}\mathrm{AKE}_{\mathcal{M}}(k)$ *is negligible in k.*

This security definition is stronger than CK-security [2] and it simultaneously captures all the known desirable security properties for authenticated key exchange including resistance to key-compromise impersonation attacks, weak perfect forward secrecy, and resilience to the leakage of ephemeral private keys.

2.7 Key-Encapsulation Mechanism (KEM)

A key encapsulation mechanism (KEM) scheme is the triple of algorithms, $\Sigma = (\mathsf{K}, \mathsf{E}, \mathsf{D})$, where

1. K, the key generation algorithm, is a probabilistic polynomial time (PPT) algorithm that takes a security parameter $k \in \mathbb{N}$ (provided in unary) and returns a pair (pk, sk) of matching public and secret keys.
2. E, the key encryption algorithm, is a PPT algorithm that takes as input public key pk and outputs a key/ciphertext pair (K^*, C^*).
3. D, the decryption algorithm, is a deterministic polynomial time algorithm that takes as input secret key sk and ciphertext C^*, and outputs key K^* or $\perp$ ($\perp$ means that the ciphertext is invalid).

We require that for all (pk, sk) output by key generation algorithm K and for all (K^*, C^*) output by key encryption algorithm $\mathsf{E}(pk)$, $\mathsf{D}(sk, C^*) = K^*$ holds. Here, the length of the key, $|K^*|$, is specified by $l(k)$, where k is the security parameter.

Let $\mathcal{A}$ be an adversary. The attack game is defined in terms of an interactive computation between adversary $\mathcal{A}$ and its challenger, $\mathcal{C}$. The challenger $\mathcal{C}$ responds to the oracle queries made by $\mathcal{A}$. We now describe the attack game (IND-CCA2 game) used to define security against adaptive chosen ciphertext attacks (IND-CCA2).

1. The challenger $\mathcal{C}$ generates a pair of keys, $(pk, sk) \stackrel{\mathsf{R}}{\leftarrow} \mathsf{K}(1^k)$ and gives pk to adversary $\mathcal{A}$.
2. Repeat the following procedure $q_1(k)$ times, for $i = 1, \ldots, q_1(k)$, where $q_1(\cdot)$ is a polynomial. $\mathcal{A}$ submits string C_i to a decryption oracle, DO (in $\mathcal{C}$), and DO returns $\mathsf{D}_{sk}(C_i)$ to $\mathcal{A}$.
3. $\mathcal{A}$ submits the encryption query to $\mathcal{C}$. The encryption oracle, EO, in $\mathcal{C}$ selects $b^* \stackrel{\mathsf{U}}{\leftarrow} \{0, 1\}$ and computes $(C^*, K^*) \leftarrow \mathsf{E}(pk)$ and returns (C^*, K^*) to $\mathcal{A}$ if $b^* = 0$ and (C^*, R^*) if $b^* = 1$, where $R^* \stackrel{\mathsf{U}}{\leftarrow} \{0, 1\}^{|K^*|}$ (C^* is called "target ciphertext").
4. Repeat the following procedure $q_2(k)$ times, for $j = q_1(k) + 1, \ldots, q_1(k) + q_2(k)$, where $q_2(\cdot)$ is a polynomial. $\mathcal{A}$ submits string C_j to a decryption oracle, DO (in $\mathcal{C}$), subject only to the restriction that a submitted text C_j is not identical to C^*. DO returns $\mathsf{D}_{sk}(C_j)$ to $\mathcal{A}$.
5. $\mathcal{A}$ outputs $b \in \{0, 1\}$.

We define the IND-CCA2 advantage of $\mathcal{A}$, $\mathsf{Adv}\mathrm{KEM}^{\text{IND-CCA2}}_{\mathcal{A}}(k) \leftarrow |\Pr[b = b^*] - 1/2|$ in the above attack game.

We say that a KEM scheme is IND-CCA2-secure (secure against adaptive chosen ciphertext attacks) if for any probabilistic polynomial-time (PPT) adversary $\mathcal{A}$, $\mathsf{Adv}\mathrm{KEM}^{\text{IND-CCA2}}_{\mathcal{A}}(k)$ is negligible in k.

3 The Proposed AKE Protocol

3.1 Protocol

Let $k \in \mathbb{N}$ be a security parameter, $\mathbb{G} \stackrel{\mathsf{U}}{\leftarrow} \{\mathbb{G}\}_k$ be a group with security parameter k, and $(g_1, g_2) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{G}^2$, where the order of $\mathbb{G}$ is prime p and $|p| = k$. Let H be a TCR hash function family, $\hat{\mathsf{F}}$ and $\tilde{\mathsf{F}}$ be PRF families, and F be a πPRF family with index $\{(I_{\mathbb{G}}, f_{\mathbb{G}})\}_{\mathbb{G} \in \{\mathbb{G}\}_k, k \in \mathbb{N}}$, where $I_{\mathbb{G}} \leftarrow \{(V, W, d) \mid (V, W, d) \in \mathbb{G}^2 \times \mathbb{Z}_p\}$ and $f_{\mathbb{G}} : (V, W, d) \mapsto V^{r_1 + dr_2} W$ with $(r_1, r_2) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$.

$(\mathbb{G}, g_1, g_2)$, H, F, $\tilde{\mathsf{F}}$ and $\hat{\mathsf{F}}$ are the system parameters common among all users of the proposed AKE protocol (although $\tilde{\mathsf{F}}$ and $\hat{\mathsf{F}}$ can be set privately by each party). We assume that the system parameters are selected by a trusted third party.

Party $\mathcal{A}$'s static private key is $(a_0, a_1, a_2, a_3, a_4) \stackrel{\mathsf{U}}{\leftarrow} (\mathbb{Z}_p)^5$ and $\mathcal{A}$'s static public key is $A_1 \leftarrow g_1^{a_1} g_2^{a_2}$, $A_2 \leftarrow g_1^{a_3} g_2^{a_4}$. $h_A \stackrel{\mathsf{R}}{\leftarrow} \mathsf{KH}_k$ indexes a TCR hash function $H_A \leftarrow \mathsf{H}^{k, \mathcal{D}_H, \mathcal{R}_H}_{h_A}$, where $\mathcal{D}_H \leftarrow \Pi_k \times \mathbb{G}^4$, $\mathcal{R}_H \leftarrow \mathbb{Z}_p$ and Π_k denotes the space of possible certificates for static public keys.

Similarly, Party $\mathcal{B}$'s static private key is $(b_0, b_1, b_2, b_3, b_4) \stackrel{\mathsf{U}}{\leftarrow} (\mathbb{Z}_p)^5$ and $\mathcal{B}$'s static public key is $B_1 \leftarrow g_1^{b_1} g_2^{b_2}$, $B_2 \leftarrow g_1^{b_3} g_2^{b_4}$. $h_B \stackrel{\mathsf{R}}{\leftarrow} \mathsf{KH}_k$ indexes a TCR hash function $H_B \leftarrow \mathsf{H}^{k, \mathcal{D}_H, \mathcal{R}_H}_{h_B}$.

$\mathcal{A}$ and $\mathcal{B}$ set πPRF and PRFs $F \leftarrow \mathsf{F}^{k, \Sigma_{\mathsf{F}}, \mathcal{D}_{\mathsf{F}}, \mathcal{R}_{\mathsf{F}}}$, $\tilde{F} \leftarrow \tilde{\mathsf{F}}^{k, \Sigma_{\tilde{\mathsf{F}}}, \mathcal{D}_{\tilde{\mathsf{F}}}, \mathcal{R}_{\tilde{\mathsf{F}}}}$ and $\hat{F} \leftarrow \hat{\mathsf{F}}^{k, \Sigma_{\hat{\mathsf{F}}}, \mathcal{D}_{\hat{\mathsf{F}}}, \mathcal{R}_{\hat{\mathsf{F}}}}$, where $\Sigma_{\mathsf{F}} \leftarrow \mathbb{G}$, $\mathcal{D}_{\mathsf{F}} \leftarrow (\Pi_k)^2 \times \mathbb{G}^{10}$, $\mathcal{R}_{\mathsf{F}} \leftarrow \{0,1\}^k$, $\Sigma_{\tilde{\mathsf{F}}} \leftarrow \mathbb{Z}_p$, $\mathcal{D}_{\tilde{\mathsf{F}}} \leftarrow \{0,1\}^k$, $\mathcal{R}_{\tilde{\mathsf{F}}} \leftarrow (\mathbb{Z}_p)^2$, $\Sigma_{\hat{\mathsf{F}}} \leftarrow \{0,1\}^k$, $\mathcal{D}_{\hat{\mathsf{F}}} \leftarrow \{0,1\}^k$, and $\mathcal{R}_{\hat{\mathsf{F}}} \leftarrow (\mathbb{Z}_p)^2$.

To establish a session key with party $\mathcal{B}$, party $\mathcal{A}$ performs the following procedure.

1. Select an ephemeral private key $(\tilde{x}_1, \tilde{x}_2) \stackrel{\mathsf{U}}{\leftarrow} \{0,1\}^k \times \{0,1\}^k$.
2. Compute $\tilde{a} \leftarrow \sum_{i=0}^{4} a_i \bmod p$, $(x, x_3) \leftarrow \hat{F}_{\tilde{x}_1}(1^k) + \tilde{F}_{\tilde{a}}(\tilde{x}_2) \bmod p$ (as two-dimensional vectors) and the ephemeral public key $(X_1 \leftarrow g_1^x, X_2 \leftarrow g_2^x, X_3 \leftarrow g_1^{x_3})$. Note that the value of (x, x_3) (and $\tilde{a}$) is only computed in a computation process of the ephemeral public key from ephemeral and static private keys.
3. Erase (x, x_3) and the whole computation history of the ephemeral public key.
4. Send $(\hat{B}, \hat{A}, X_1, X_2, X_3)$ to $\mathcal{B}$.

Upon receiving $(\hat{B}, \hat{A}, X_1, X_2, X_3)$, party $\mathcal{B}$ verifies that $(X_1, X_2, X_3) \in \mathbb{G}^3$. If so, perform the following procedure.

1. Select an ephemeral private key $(\tilde{y}_1, \tilde{y}_2) \stackrel{\mathsf{U}}{\leftarrow} \{0,1\}^k \times \{0,1\}^k$.

2. Compute $\tilde{b} \leftarrow \sum_{i=0}^{4} b_i \bmod p$, $(y, y_3) \leftarrow \hat{F}_{\tilde{y}_1}(1^k) + \tilde{F}_{\tilde{b}}(\tilde{y}_2) \bmod p$ (as two-dimensional vectors) and the ephemeral public key $(Y_1 \leftarrow g_1^y, Y_2 \leftarrow g_2^y, Y_3 \leftarrow g_1^{y_3})$.
3. Erase (y, y_3) and the whole computation history of the ephemeral public key.
4. Send $(\hat{A}, \hat{B}, X_1, X_2, X_3, Y_1, Y_2, Y_3)$ to $\mathcal{A}$.

Upon receiving $(\hat{A}, \hat{B}, X_1, X_2, X_3, Y_1, Y_2, Y_3)$, party $\mathcal{A}$ checks if he sent $(\hat{B}, \hat{A}, X_1, X_2, X_3)$ to $\mathcal{B}$. If so, $\mathcal{A}$ verifies that $(Y_1, Y_2, Y_3) \in \mathbb{G}^3$.

To compute the session key, $\mathcal{A}$ computes $\sigma_A \leftarrow Y_1^{a_1+ca_3} Y_2^{a_2+ca_4} Y_3^{x_3} B_1^x B_2^{dx}$, and $\mathcal{B}$ computes $\sigma_B \leftarrow X_1^{b_1+db_3} X_2^{b_2+db_4} X_3^{y_3} A_1^y A_2^{cy}$, where $c \leftarrow H_A(\hat{A}, Y_1, Y_2)$ and $d \leftarrow H_B(\hat{B}, X_1, X_2)$. If they are correctly computed, $\sigma \leftarrow \sigma_A(= \sigma_B)$. The session key is $K \leftarrow F_\sigma(\mathsf{sid})$, where $\mathsf{sid} \leftarrow (\hat{A}, \hat{B}, X_1, X_2, X_3, Y_1, Y_2, Y_3)$.

3.2 Security

Theorem 1. *The proposed AKE protocol is secure (in the sense of Definition 2) if the DDH assumption holds for* $\{\mathbb{G}\}_{k\in\mathbb{N}}$, H *is a TCR hash function family,* $\tilde{\mathsf{F}}$ *and* $\hat{\mathsf{F}}$ *are PRF families, and* F *is a* π*PRF family with index* $\{(I_\mathbb{G}, f_\mathbb{G})\}_{\mathbb{G}\in\{\mathbb{G}\}_k, k\in\mathbb{N}}$*, where* $I_\mathbb{G} \leftarrow \{(V, W, d) \mid (V, W, d) \in \mathbb{G}^2 \times \mathbb{Z}_p\}$ *and* $f_\mathbb{G} : (V, W, d) \mapsto V^{r_1+dr_2} W$ *with* $(r_1, r_2) \xleftarrow{\mathsf{U}} \mathbb{Z}_p^2$.

Proof. It is obvious that the first condition of Definition 2 holds.

We will prove that the second condition of Definition 2 holds under the assumptions.

Let sid^* be the target session chosen by adversary $\mathcal{M}$, $\mathcal{A}$ be the owner of the session sid^* and $\mathcal{B}$ be the peer. Let sid^* be $(\hat{A}, \hat{B}, X_1^*, X_2^*, X_3^*, Y_1^*, Y_2^*, Y_3^*)$, where $\hat{A}$ includes (A_1,A_2), $\hat{B}$ includes $(B_1,,B_2)$, $A_1 \leftarrow g_1^{a_1^*} g_2^{a_2^*}$, $A_2 \leftarrow g_1^{a_3^*} g_2^{a_4^*}$, $B_1 \leftarrow g_1^{b_1^*} g_2^{b_2^*}$, $B_2 \leftarrow g_1^{b_3^*} g_2^{b_4^*}$, $X_1^* \leftarrow g_1^{x^*}, X_2^* \leftarrow g_2^{x^*}, X_3^* \leftarrow g_1^{x_3^*}$ $Y_1^* \leftarrow g_1^{y^*}, Y_2^* \leftarrow g_2^{y^*}, Y_3^* \leftarrow g_1^{y_3^*}$.

We will evaluate the advantage, $\mathsf{Adv}\mathrm{AKE}_\mathcal{M}(k)$, in the following two disjoint cases (which cover the whole):

- Case 1: there exists a matching session, $\overline{\mathsf{sid}^*}$, of target session sid^*,
- Case 2: there exists no matching session of target session sid^*.

<u>**Case 1:**</u>

To evaluate $\mathsf{Adv}\mathrm{AKE}_\mathcal{M}(k)$ in Case 1, we consider five games, $\mathbf{G}_0^{(1)}$, $\mathbf{G}_1^{(1)}$, $\mathbf{G}_2^{(1)}$, $\mathbf{G}_3^{(1)}$, $\mathbf{G}_4^{(1)}$ as follows:

Game $\mathbf{G}_0^{(1)}$. This is the original eCK game with adversary $\mathcal{M}$ to define $\mathsf{Adv}\mathrm{AKE}_\mathcal{M}(k)$ in Case 1.

Game $\mathbf{G}_1^{(1)}$. This is a *local* eCK game with an adversary $\mathcal{M}_1$ that is is reduced from game $\mathbf{G}_0^{(1)}$ with adversary $\mathcal{M}$. In the local eCK game in Case 1, $\mathcal{M}_1$ activates only two parties (say $\mathcal{A}$ and $\mathcal{B}$) (except dishonest parties) and only two sessions, the target session and the matching session (say $(\hat{A}, \hat{B}, X_1^*, X_2^*, X_3^*, Y_1^*, Y_2^*, Y_3^*)$ and $(\hat{B}, \hat{A}, Y_1^*, Y_2^*, Y_3^*, X_1^*, X_2^*, X_3^*)$) (except sessions with dishonest parties).

$\mathcal{A}$		$\mathcal{B}$
$(a_0, a_1, a_2, a_3, a_4) \xleftarrow{\mathsf{U}} (\mathbb{Z}_p)^5$		$(b_0, b_1, b_2, b_3, b_4) \xleftarrow{\mathsf{U}} (\mathbb{Z}_p)^5$
$A_1 \leftarrow g_1^{a_1} g_2^{a_2}, A_2 \leftarrow g_1^{a_3} g_2^{a_4},$		$B_1 \leftarrow g_1^{b_1} g_2^{b_2}, B_2 \leftarrow g_1^{b_3} g_2^{b_4},$
h_A		h_B
$(\tilde{x}_1, \tilde{x}_2) \xleftarrow{\mathsf{U}} \{0,1\}^k \times \{0,1\}^k$		
$(x, x_3) \leftarrow \hat{F}_{\tilde{x}_1}(1^k)$		
$\quad + \tilde{F}_{\tilde{a}}(\tilde{x}_2) \bmod p$		
$(\tilde{a} \leftarrow \sum_{i=0}^{4} a_i \bmod p)$		
$X_1 \leftarrow g_1^x, X_2 \leftarrow g_2^x,$		
$X_3 \leftarrow g_1^{x_3}$		
	$\xrightarrow{(\hat{B}, \hat{A}, X_1, X_2, X_3)}$	$(X_1, X_2, X_3) \in \mathbb{G}^3?$
		$(\tilde{y}_1, \tilde{y}_2) \xleftarrow{\mathsf{U}} \{0,1\}^k \times \{0,1\}^k$
		$(y, y_3) \leftarrow \hat{F}_{\tilde{y}_1}(1^k)$
		$\quad + \tilde{F}_{\tilde{b}}(\tilde{y}_2) \bmod p$
		$(\tilde{b} \leftarrow \sum_{i=0}^{4} b_i \bmod p)$
		$Y_1 \leftarrow g_1^y, Y_2 \leftarrow g_2^y,$
		$Y_3 \leftarrow g_1^{y_3}$
$(Y_1, Y_2, Y_3) \in \mathbb{G}^3?$	$\xleftarrow{(\hat{A}, \hat{B}, X_1, X_2, X_3, Y_1, Y_2, Y_3)}$	
$c \leftarrow H_A(\hat{A}, Y_1, Y_2)$		$c \leftarrow H_A(\hat{A}, Y_1, Y_2)$
$d \leftarrow H_B(\hat{B}, X_1, X_2)$		$d \leftarrow H_B(\hat{B}, X_1, X_2)$
$\sigma \leftarrow Y_1^{a_1 + ca_3} Y_2^{a_2 + ca_4} \cdot$		$\sigma \leftarrow X_1^{b_1 + db_3} X_2^{b_2 + db_4} \cdot$
$\quad Y_3^{x_3} B_1^x B_2^{dx}$		$\quad X_3^{y_3} A_1^y A_2^{cy}$
$K \leftarrow F_\sigma(\mathsf{sid})$		$K \leftarrow F_\sigma(\mathsf{sid})$

Here, $\mathsf{sid} \leftarrow (\hat{A}, \hat{B}, X_1, X_2, X_3, Y_1, Y_2, Y_3)$. Note that $(A_1, A_2, B_1, B_2) \in \mathbb{G}^4$ is confirmed indirectly through the certificates.

Fig. 1. The Proposed AKE

Game $\mathbf{G}_2^{(1)}$. We modify game $\mathbf{G}_1^{(1)}$ to game $\mathbf{G}_2^{(1)}$ by changing PRFs $\tilde{F}_{\tilde{a}^*}$, $\tilde{F}_{\tilde{b}^*}$, $\hat{F}_{\tilde{x}_1^*}$ and $\hat{F}_{\tilde{y}_1^*}$ of the target and matching sessions to random functions.

Game $\mathbf{G}_3^{(1)}$. We modify game $\mathbf{G}_2^{(1)}$ to game $\mathbf{G}_3^{(1)}$ by changing the value of $(Y_3^*)^{x_3^*} = (X_3^*)^{y_3^*}$ to a random element $\delta \xleftarrow{\mathsf{U}} \mathbb{G}$.

Game $\mathbf{G}_4^{(1)}$. We modify game $\mathbf{G}_3^{(1)}$ to game $\mathbf{G}_4^{(1)}$ by changing PRF F_{σ^*} to a random function. Note that the requirement of PRF for F_{σ^*} is sufficient here (πPRF is not necessary).

Let $\mathsf{Adv}_0^{(1)}$ be the eCK advantage of $\mathcal{M}$ in game $\mathbf{G}_0^{(1)}$ (i.e., $\mathsf{Adv}\mathrm{AKE}_{\mathcal{M}}(k)$ in Case 1). Let $\mathsf{Adv}_i^{(1)}$ ($i = 1, 2, 3, 4$) be the eCK advantage of $\mathcal{M}_1$ in game $\mathbf{G}_i^{(1)}$.

We will then evaluate the relations between pairs of the advantages.

Claim 1. *For any adversary $\mathcal{M}$ in game $\mathbf{G}_0^{(1)}$ and any (correctly set-up) local eCK game $\mathbf{G}_1^{(1)}$, there exists an adversary, $\mathcal{M}_1$, for the local eCK game, and a machine $\mathcal{M}_2$ whose running times are at most that of $\mathcal{M}$, such that*

$$\mathsf{Adv}_0^{(1)} < 4n(k)^2 s(k) \cdot \mathsf{Adv}_1^{(1)} + s(k) \cdot \mathsf{Adv}\mathrm{PRF}_{\tilde{\mathsf{F}},\mathcal{M}_2}(k)$$

where $\mathcal{M}$ activates at most $s(k)$ sessions.

Proof. Let's suppose that $\mathcal{M}$ activates at most $n(k)$ honest parties. Given an adversary $\mathcal{M}$ in game $\mathbf{G}_0^{(1)}$ and a (correctly set-up) local eCK game with two parties, ($\mathcal{A}$ and $\mathcal{B}$), we construct $\mathcal{M}_1$ as follows: First, $\mathcal{M}_1$ randomly establishes $(n(k) - 2)$ honest parties correctly in addition to $\mathcal{A}$ and $\mathcal{B}$. $\mathcal{M}_1$ then simulates the eCK game for the $n(k)$ honest parties (including $\mathcal{A}$ and $\mathcal{B}$) with $\mathcal{M}$. $\mathcal{M}_1$ randomly guesses the target session whose owner and peer are $\mathcal{A}$ and $\mathcal{B}$.

$\mathcal{M}_1$'s simulation is executed as follows:

1. $\mathcal{M}_1$ selects $\alpha \leftarrow (\alpha_1, \alpha_2) \overset{\mathsf{U}}{\leftarrow} \{0,1\}^2$. Intuitively, $\alpha_1 =$ '0' means $\mathcal{M}_1$'s guess that $\mathcal{M}$ issues no ephemeral key reveal query on $\mathcal{A}$ for the guessed target session, and $\alpha_1 =$ '1' means the opposite. $\alpha_2 =$ '0' means $\mathcal{M}_1$'s guess that $\mathcal{M}$ issues no ephemeral key reveal query on $\mathcal{B}$ for the matching session of the guessed target session, and $\alpha_2 =$ '1' means the opposite. Due to the conditions of a target session (or a fresh session), if $\mathcal{M}$ issues an ephemeral key reveal query for a target session, $\mathcal{M}$ cannot issue the static key reveal query on the owner of the target session.
2. If α is '00', $\mathcal{M}_1$ issues static key reveal queries on $\mathcal{A}$ and $\mathcal{B}$ in the beginning of the game, and then starts the simulation of the eCK game with $\mathcal{M}$.
   In the simulation,
   (a) $\mathcal{M}_1$ simulates the sessions of the established $(n(k) - 2)$ honest parties correctly.
   (b) If a session of $\mathcal{A}$ or $\mathcal{B}$ is not the guessed target session nor the matching session, $\mathcal{M}_1$ correctly simulates the session (i.e. selects an ephemeral private key and computes ephemeral public key correctly by using the static and ephemeral private keys).
   (c) If a session of $\mathcal{A}$ or $\mathcal{B}$ is the guessed target session or the matching session, execute the local eCK game.

   If $\mathcal{M}_1$'s guess is incorrect (i.e., $\mathcal{M}$ does not select the guessed target session as the target session or $\mathcal{M}$ issues an ephemeral key reveal query for the guessed target or the matching session), $\mathcal{M}_1$ aborts this game (game $\mathbf{G}_1^{(1)}$).
3. If α is '01', $\mathcal{M}_1$ issues a static key reveal query on $\mathcal{A}$ in the beginning of the game, and then starts the simulation of the eCK game with $\mathcal{M}$.
   In the simulation,
   (a) $\mathcal{M}_1$ simulates the sessions of the established $(n(k) - 2)$ honest parties correctly.
   (b) If a session of $\mathcal{A}$ is not the guessed target session, $\mathcal{M}_1$ correctly simulates the session (i.e. selects an ephemeral private key and computes ephemeral public key correctly by using the static and ephemeral private keys).

(c) If a session of $\mathcal{B}$ is not the matching session of the guessed target session, $\mathcal{M}_1$ selects $(y, y_3) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$ and computes $Y_1 \leftarrow g_1^y$, $Y_2 \leftarrow g_2^y$ and $Y_3 \leftarrow g_1^{y_3}$.

(d) If a session of $\mathcal{A}$ or $\mathcal{B}$ is the guessed target session or the matching session, execute the local eCK game.

If $\mathcal{M}_1$'s guess is incorrect (i.e., $\mathcal{M}$ does not select the guessed target session or $\mathcal{M}$ does not issue an ephemeral key reveal query for the matching session or issues an ephemeral key reveal query for the guessed target session), $\mathcal{M}_1$ aborts this game.

4. If α is '10', $\mathcal{M}_1$ issues a static key reveal query on $\mathcal{B}$ in the beginning of the game, and then starts the simulation of the eCK game with $\mathcal{M}$.
In the simulation,
(a) $\mathcal{M}_1$ simulates the sessions of the established $(n(k) - 2)$ honest parties correctly.
(b) If a session of $\mathcal{B}$ is not the matching session of the guessed target session, $\mathcal{M}_1$ correctly simulates the session (i.e. selects an ephemeral private key and computes ephemeral public key correctly by using the static and ephemeral private keys).
(c) If a session of $\mathcal{A}$ is not the guessed target session, $\mathcal{M}_1$ selects $(x, x_3) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$ and computes $X_1 \leftarrow g_1^x$, $X_2 \leftarrow g_2^x$ and $X_3 \leftarrow g_1^{x_3}$.
(d) If a session of $\mathcal{A}$ or $\mathcal{B}$ is the guessed target session or the matching session, execute the local eCK game.

If $\mathcal{M}_1$'s guess is incorrect (i.e., $\mathcal{M}$ does not select the guessed target session or $\mathcal{M}$ does not issue an ephemeral key reveal query for the guessed target session or issues an ephemeral key reveal query for the matching session), $\mathcal{M}_1$ aborts this game.

5. If α is '11', $\mathcal{M}_1$ starts the simulation of the eCK game with $\mathcal{M}$.
In the simulation,
(a) $\mathcal{M}_1$ simulates the sessions of the established $(n(k) - 2)$ honest parties correctly.
(b) If a session of $\mathcal{A}$ or $\mathcal{B}$ is not the guessed target session nor the matching session, $\mathcal{M}_1$ selects $(x, x_3) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$ and computes $X_1 \leftarrow g_1^x$, $X_2 \leftarrow g_2^x$ and $X_3 \leftarrow g_1^{x_3}$ (or selects $(y, y_3) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$ and computes $Y_1 \leftarrow g_1^y$, $Y_2 \leftarrow g_2^y$ and $Y_3 \leftarrow g_1^{y_3}$).
(c) If a session of $\mathcal{A}$ or $\mathcal{B}$ is the guessed target session or the matching session, execute the local eCK game.
6. $\mathcal{M}_1$ finally outputs the output of $\mathcal{M}$, unless $\mathcal{M}_1$ aborts the game.

If $\mathcal{M}_1$'s guess (on the target session and α) is correct and $\alpha = 00$, $\mathcal{M}_1$'s advantage in this simulation is exactly equivalent to $\mathcal{M}_0$'s advantage in game $\mathbf{G}_0^{(1)}$.

If $\mathcal{M}_1$'s guess (on the target session and α) is correct and $\alpha \in \{01, 10, 11\}$, the difference between $\mathcal{M}_1$'s advantage in this simulation and $\mathcal{M}_0$'s advantage in game $\mathbf{G}_0^{(1)}$ can be evaluated as follows:

We now assume a PRF security test environment for $\tilde{\mathsf{F}}$, where adversary $\mathcal{M}_2$ is allowed to access to two oracles, which are $(\tilde{F}_{\delta_1}, \tilde{F}_{\delta_2})$ ($(\delta_1, \delta_2) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$) or two random functions (RF_1, RF_2).

We then construct $\mathcal{M}_2$ as follows: $\mathcal{M}_2$ simulates the sessions of $\mathcal{A}$ and $\mathcal{B}$ correctly except the computation of $\tilde{F}_{\tilde{a}}$ and $\tilde{F}_{\tilde{b}}$ of $\mathcal{A}$ and $\mathcal{B}$, where in place of $\mathcal{M}_1$'s selecting

$(x, x_3) \xleftarrow{\mathsf{U}} \mathbb{Z}_p^2$ and/or $(y, y_3) \xleftarrow{\mathsf{U}} \mathbb{Z}_p^2$ (in cases of $\alpha \in \{01, 10, 11\}$), $\mathcal{M}_2$ sends the related queries to the oracles. Finally $\mathcal{M}_2$ outputs 1 iff $\mathcal{M}_1$ correctly guesses b^* (i.e., $\mathcal{M}_1$'s output b is equivalent to b^* in (Definition 2 of) game $\mathbf{G}_0^{(1)}$).

If the oracles are $(\tilde{F}_{\delta_1}, \tilde{F}_{\delta_2})$, then the simulation with the oracle queries is equivalent to game $\mathbf{G}_0^{(1)}$, since the distribution of $\tilde{a}^*$ and $\tilde{b}^*$ are independent and uniform over $\mathbb{Z}_p$. Otherwise, it is equivalent to $\mathcal{M}_1$'s simulation described above under the condition that $\mathcal{M}_1$'s guess is correct. The number of calls to the oracles is bounded by $s(k)$ in all cases of $\alpha \in \{01, 10, 11\}$, So, applying the hybrid argument, (where M_2 sets up the i-th step of the hybrid argument for $i = 1, \ldots, s(k)$), we obtain

$$|\mathsf{Adv}_0^{(1)} - \mathsf{Adv}_1^{(1)}[\mathsf{CorrGuess}]| < s(k) \cdot \mathsf{AdvPRF}_{\tilde{\mathsf{F}}, \mathcal{M}_2}(k),$$

where $\mathsf{Adv}_1^{(1)}[\mathsf{CorrGuess}]$ is the advantage $\mathsf{Adv}_1^{(1)}$ under the condition that $\mathcal{M}_1$'s guess is correct.

Since the probability that $\mathcal{M}_1$'s guess on the target session is correct is at least $1/(n(k)^2 s(k))$ and the probability that $\mathcal{M}_1$'s guess is correct on α is 1/4,

$$1/(4n(k)^2 s(k)) \cdot (\mathsf{Adv}_0^{(1)} - s(k) \cdot \mathsf{AdvPRF}_{\tilde{\mathsf{F}}, \mathcal{M}_2}(k)) < \mathsf{Adv}_1^{(1)}.$$

□

Claim 2. *There exists a probabilistic machine $\mathcal{M}_3$, whose running time is at most that of $\mathcal{M}$, such that*

$$|\mathsf{Adv}_1^{(1)} - \mathsf{Adv}_2^{(1)}| \leq 2 \cdot \max\{\mathsf{AdvPRF}_{\tilde{\mathsf{F}}, \mathcal{M}_3}(k), \mathsf{AdvPRF}_{\hat{\mathsf{F}}, \mathcal{M}_3}(k)\}.$$

Proof. We now assume PRF security test environments for $\tilde{\mathsf{F}}$ and $\hat{\mathsf{F}}$, where adversary $\mathcal{M}_3$ is allowed to access to four oracles, which are $(\tilde{\mathsf{F}}_{\delta_1}, \tilde{\mathsf{F}}_{\delta_2}, \hat{\mathsf{F}}_{\xi_1}, \hat{\mathsf{F}}_{\xi_2})$ $((\delta_1, \delta_2) \xleftarrow{\mathsf{U}} \mathbb{Z}_p^2$, $(\xi_1, \xi_2) \xleftarrow{\mathsf{U}} \{0, 1\}^{2k})$ or four random functions (RF_1, RF_2, RF_3, RF_4).

We construct $\mathcal{M}_3$ as follows: $\mathcal{M}_3$ sets up the parameters of game $\mathbf{G}_1^{(1)}$ for two parties, $\mathcal{A}$ and $\mathcal{B}$, and the target and matching sessions correctly and simulates the game with adversary $\mathcal{M}_1$ except the computation of $\tilde{\mathsf{F}}_{\tilde{a}^*}(\tilde{x}_2^*)$, $\tilde{\mathsf{F}}_{\tilde{b}^*}(\tilde{y}_2^*)$, $\hat{\mathsf{F}}_{\tilde{x}_1^*}(1^k)$ and $\hat{\mathsf{F}}_{\tilde{y}_1^*}(1^k)$, where $\mathcal{M}_3$ accesses to the oracles and sets the returned values as the function values. Finally $\mathcal{M}_3$ outputs 1 iff $\mathcal{M}_1$ correctly guesses b^* (i.e., $\mathcal{M}_1$'s output b is equivalent to b^* in (Definition 2 of) game $\mathbf{G}_1^{(1)}$).

If the oracle is $\tilde{\mathsf{F}}$ and $\hat{\mathsf{F}}$, the simulation is equivalent to game $\mathbf{G}_1^{(1)}$. Otherwise, the simulation is equivalent to game $\mathbf{G}_2^{(1)}$.

Since both the static and ephemeral keys of the target (matching) session are not revealed at the same time, we obtain

$$|\mathsf{Adv}_1^{(1)} - \mathsf{Adv}_2^{(1)}| \leq 2 \cdot \max\{\mathsf{AdvPRF}_{\tilde{\mathsf{F}}, \mathcal{M}_3}(k), \mathsf{AdvPRF}_{\hat{\mathsf{F}}, \mathcal{M}_3}(k)\}.$$

□

Claim 3. *There exists a probabilistic machine $\mathcal{M}_4$, whose running time is at most that of $\mathcal{M}$, such that*

$$|\mathsf{Adv}_2^{(1)} - \mathsf{Adv}_3^{(1)}| = \mathsf{Adv}\mathrm{DDH}_{\mathcal{M}_4}(k).$$

Proof. Given a DDH problem $\rho \leftarrow (\mathbb{G}, U, V, W, Z)$, where $\rho \stackrel{\mathsf{U}}{\leftarrow} \mathbb{D}(k)$ or $\rho \stackrel{\mathsf{U}}{\leftarrow} \mathbb{R}(k)$, we construct its adversary $\mathcal{M}_4$ using $\mathcal{M}_1$ in game $\mathbf{G}_2^{(1)}$ as follows:

$\mathcal{M}_4$ sets up the parameters of game $\mathbf{G}_2^{(1)}$ for two parties, $\mathcal{A}$ and $\mathcal{B}$, correctly and simulates the game with adversary $\mathcal{M}_1$ except the computation of g_1, X_3^*, Y_3^* and $(Y_3^*)^{x_3^*}(= (X_3^*)^{y_3^*})$.

For the computation, $\mathcal{M}_4$ sets $g_1 \leftarrow U$, $X_3^* \leftarrow V$, $Y_3^* \leftarrow W$, and sets Z as the specified value of $(Y_3^*)^{x_3^*}(= (X_3^*)^{y_3^*})$. (Note that the simulation of the other values can be perfectly executed with using g_1, X_3^*, Y_3^* and $(Y_3^*)^{x_3^*}(= (X_3^*)^{y_3^*})$.)

Finally $\mathcal{M}_4$ outputs 1 iff $\mathcal{M}_1$ correctly guesses b^* (i.e., $\mathcal{M}_1$'s output b is equivalent to b^* in (Definition 2 of) game $\mathbf{G}_2^{(1)}$).

If $\rho \stackrel{\mathsf{U}}{\leftarrow} \mathbb{D}(k)$, the advantage of $\mathcal{M}_1$ in this simulation is equivalent to that in game $\mathbf{G}_2^{(1)}$, $\mathsf{Adv}_2^{(1)}$. Otherwise, the advantage of $\mathcal{M}_1$ in this simulation is equivalent to that in game $\mathbf{G}_3^{(1)}$, $\mathsf{Adv}_3^{(1)}$.

Therefore, $|\mathsf{Adv}_2^{(1)} - \mathsf{Adv}_3^{(1)}| = \mathsf{Adv}\mathrm{DDH}_{\mathcal{M}_4}(k)$. □

Claim 4. *There exists a probabilistic machine $\mathcal{M}_5$, whose running time is at most that of $\mathcal{M}$, such that*

$$|\mathsf{Adv}_3^{(1)} - \mathsf{Adv}_4^{(1)}| \leq \mathsf{Adv}\mathrm{PRF}_{\mathsf{F},\mathcal{M}_5}(k).$$

Proof. Given a PRF security test environment for F, where an adversary is allowed to access an oracle, F_γ ($\gamma \stackrel{\mathsf{U}}{\leftarrow} \mathbb{G}$) or a random function RF, and tries to distinguish them, we construct its adversary $\mathcal{M}_5$ using $\mathcal{M}_1$ in game $\mathbf{G}_3^{(1)}$ as follows:

$\mathcal{M}_5$ sets up the parameters of game $\mathbf{G}_3^{(1)}$ for two parties, $\mathcal{A}$ and $\mathcal{B}$, correctly and simulates the game with adversary $\mathcal{M}_1$ except the computation of $K \leftarrow F_{\sigma^*}(\mathsf{sid}^*)$, where $\mathcal{M}_5$ sends sid^* to the oracle and sets the value returned from the oracle as K. Finally $\mathcal{M}_5$ outputs 1 iff $\mathcal{M}_1$ correctly guesses b^* (i.e., $\mathcal{M}_1$'s output b is equivalent to b^* in (Definition 2 of) game $\mathbf{G}_3^{(1)}$).

If the oracle is F_γ, the returned value from the oracle is perfectly indistinguishable from that of $F_{\sigma^*}(\mathsf{sid})$, since the value of σ^* in game $\mathbf{G}_3^{(1)}$ is uniform and independent. Then, the advantage of $\mathcal{M}_1$ in this simulation is equivalent to that in game $\mathbf{G}_3^{(1)}$, $\mathsf{Adv}_3^{(1)}$. Otherwise, the advantage of $\mathcal{M}_1$ in this simulation is equivalent to that in game $\mathbf{G}_4^{(1)}$, $\mathsf{Adv}_4^{(1)}$.

Therefore, $|\mathsf{Adv}_3^{(1)} - \mathsf{Adv}_4^{(1)}| \leq \mathsf{Adv}\mathrm{PRF}_{\mathsf{F},\mathcal{M}_5}(k)$. □

Summing up Claims 1 to 4 and from the fact that $\mathsf{Adv}_4^{(1)} = 0$, we obtain the following claim,

Claim 5. *Let's suppose Case 1 occurs. For any adversary $\mathcal{M}$ in the eCK game (Definition 2), there exist probabilistic machines, $\mathcal{M}_2, \mathcal{M}_3, \mathcal{M}_4$ and $\mathcal{M}_5$, whose running times are at most that of $\mathcal{M}$, such that*

$$\begin{aligned}\mathsf{Adv}\mathrm{AKE}_{\mathcal{M}}(k) \; &< 4n(k)^2 s(k) \cdot (2 \cdot \max\{\mathsf{Adv}\mathrm{PRF}_{\tilde{\mathsf{F}},\mathcal{M}_3}(k), \mathsf{Adv}\mathrm{PRF}_{\hat{\mathsf{F}},\mathcal{M}_3}(k)\} \\ &\quad + \mathsf{Adv}\mathrm{DDH}_{\mathcal{M}_4}(k) + \mathsf{Adv}\mathrm{PRF}_{\mathsf{F},\mathcal{M}_5}(k)) + s(k) \cdot \mathsf{Adv}\mathrm{PRF}_{\tilde{\mathsf{F}},\mathcal{M}_2}(k).\end{aligned}$$

Case 2:

Next, we will evaluate $\mathsf{Adv}\mathrm{AKE}_{\mathcal{M}}(k)$ in Case 2. Recall that sid^* is the target session chosen by adversary $\mathcal{M}$, $\mathcal{A}$ is the owner of the session sid^* and $\mathcal{B}$ is the peer. Let sid^* be $(\hat{A}, \hat{B}, X_1^*, X_2^*, X_3^*, Y_1^*, Y_2^*, Y_3^*)$.

In Case 2, $\mathcal{B}$ is a honest party, but does not own a session that is matching to session sid^*. Due to the conditions of a fresh session (i.e., restrictions on sid^*), $\mathcal{M}$ cannot issue a static key reveal query on $\mathcal{B}$, but $\mathcal{M}$ (or a dishonest party) can establish a session, sid_i, $(\hat{C}^{(i)}, \hat{B}, X_1^{(i)}, X_2^{(i)}, X_3^{(i)}, Y_1^{(i)}, Y_2^{(i)}, Y_3^{(i)})$, with $\mathcal{B}$ that is not the matching session of sid^* for $i = 1, \ldots, t(k)$ (i.e., $(\hat{A},\ \hat{B},\ X_1^*, X_2^*, X_3^*, Y_1^*, Y_2^*, Y_3^*) \neq (\hat{C}^{(i)},$ $\hat{B}^{(i)},\ X_1^{(i)}, X_2^{(i)}, X_3^{(i)}, Y_1^{(i)}, Y_2^{(i)}, Y_3^{(i)}))$ and can issue a session key reveal query on the session $(\hat{C}^{(i)}, \hat{B}, X_1^{(i)}, X_2^{(i)}, X_3^{(i)}, Y_1^{(i)}, Y_2^{(i)}, Y_3^{(i)})$.

We consider seven games, $\mathbf{G}_0^{(2)}$, $\mathbf{G}_1^{(2)}$, $\mathbf{G}_2^{(2)}$, $\mathbf{G}_3^{(2)}$, $\mathbf{G}_4^{(2)}$, $\mathbf{G}_5^{(2)}$ and $\mathbf{G}_6^{(2)}$, as follows:

Game $\mathbf{G}_0^{(2)}$. This is the original eCK game with adversary $\mathcal{M}$ to define $\mathsf{Adv}\mathrm{AKE}_{\mathcal{M}}(k)$ in Case 2.

Game $\mathbf{G}_1^{(2)}$. This is a *local* eCK game with adversary $\mathcal{M}_{11}$ that is reduced from the original eCK game with adversary $\mathcal{M}$. In the local eCK game in Case 2, $\mathcal{M}_{11}$ activates only two parties (e.g., $\mathcal{A}$ and $\mathcal{B}$) (except dishonest parties) and the target session (e.g., $(\hat{A}, \hat{B}, X_1^*, X_2^*, X_3^*, Y_1^*, Y_2^*, Y_3^*)$) (except sessions with dishonest parties).

Game $\mathbf{G}_2^{(2)}$. We modify game $\mathbf{G}_1^{(2)}$ to game $\mathbf{G}_2^{(2)}$ by changing PRFs $\tilde{F}_{\tilde{a}^*}$, $\tilde{F}_{\tilde{b}^*}$, $\hat{F}_{\tilde{x}_1^*}$ and $\hat{F}_{\tilde{y}_1^{(i)}}$ $(i = 1, \ldots, t(k))$ to random functions.

Game $\mathbf{G}_3^{(2)}$. We modify game $\mathbf{G}_2^{(2)}$ to game $\mathbf{G}_3^{(2)}$ by changing the value of $B_1^{x^*} B_2^{d^* x^*}$ (in the computation process of $\sigma^* \leftarrow (Y_1^*)^{a_1^* + c^* a_3^*} (Y_2^*)^{a_2^* + c^* a_4^*} (Y_3^*)^{x_3^*} B_1^{x^*} B_2^{d^* x^*}$) to $(X_1^*)^{b_1^* + d^* b_3^*} (X_2^*)^{b_2^* + d^* b_4^*}$. This change is purely conceptual.

Game $\mathbf{G}_4^{(2)}$. We modify game $\mathbf{G}_3^{(2)}$ to game $\mathbf{G}_4^{(2)}$ by changing DH tuple $(\mathbb{G}, g_1, g_2,$ $X_1^*, X_2^*) \overset{\mathsf{U}}{\leftarrow} \mathbb{D}(k)$ to a random tuple $(\mathbb{G}, g_1, g_2, X_1^*, X_2^*) \overset{\mathsf{U}}{\leftarrow} \mathbb{R}(k)$.

Game $\mathbf{G}_5^{(2)}$. We modify game $\mathbf{G}_4^{(2)}$ to game $\mathbf{G}_5^{(2)}$ by adding a special rejection rule in game $\mathbf{G}_5^{(2)}$, such that game $\mathbf{G}_5^{(2)}$ aborts if $\mathcal{M}$ (dishonest party $\mathcal{C}$) establishes a session with $\mathcal{B}$, $(\hat{C}^{(i)}, \hat{B}, X_1^{(i)}, X_2^{(i)}, X_3^{(i)}, Y_1^{(i)}, Y_2^{(i)}, Y_3^{(i)})$, issues a session key query on the session, and $H_B(\hat{B}, X_1^*, X_2^*) = H_B(\hat{B}, X_1^{(i)}, X_2^{(i)})$ and $(\hat{B}, X_1^*, X_2^*) \neq$ $(\hat{B}, X_1^{(i)}, X_2^{(i)})$ occur.

Game $\mathbf{G}_6^{(2)}$. We modify game $\mathbf{G}_5^{(2)}$ to game $\mathbf{G}_6^{(2)}$ by changing a πPRF of the target session, F_{σ^*}, to a random function.

Let $\mathsf{Adv}_0^{(2)}$ be the eCK advantage of $\mathcal{M}$ in game $\mathbf{G}_0^{(2)}$ (i.e., $\mathsf{Adv}\mathrm{AKE}_{\mathcal{M}}(k)$ in Case 2). Let $\mathsf{Adv}_i^{(2)}$ ($i = 1, 2, 3, 4, 5, 6$) be the eCK advantage of $\mathcal{M}_{11}$ in game $\mathbf{G}_i^{(2)}$.

We now proceed to evaluate the differences between pairs of the advantages.

Claim 6. *For an adversary $\mathcal{M}$ in game $\mathbf{G}_0^{(2)}$ and a (correctly set-up) local eCK game, there exists an adversary, $\mathcal{M}_{11}$, for the local eCK game, and a machine $\mathcal{M}_{12}$ whose running times are at most that of $\mathcal{M}$, such that*

$$\mathsf{Adv}_0^{(2)} < 2n(k)^2 s(k) \cdot \mathsf{Adv}_1^{(2)} + s(k) \cdot \mathsf{Adv}\mathrm{PRF}_{\tilde{\mathsf{F}},\mathcal{M}_{12}}(k)$$

where $\mathcal{M}$ activates at most $s(k)$ sessions.

Proof. The proof of this claim is similar to that of Claim 1. The only difference is for $\mathcal{M}_{11}$'s (and $\mathcal{M}_2$'s) guess on α. In Case 1, $\mathcal{B}$ owns the matching session of the target session, while $\mathcal{B}$ owns no matching session in Case 2, but has a restriction on key reveals such that $\mathcal{B}$'s static key cannot be revealed. Therefore, $\mathcal{M}_{11}$ only needs to make a one-bit guess on $\mathcal{A}$'s key reveal (static or ephemeral key reveal) to complete the simulation. We then obtain

$$1/(2n(k)^2 s(k)) \cdot (\mathsf{Adv}_0^{(2)} - s(k) \cdot \mathsf{Adv}\mathrm{PRF}_{\tilde{\mathsf{F}},\mathcal{M}_{12}}(k)) < \mathsf{Adv}_1^{(2)}.$$

□

The proof of the following claim is also similar to that of Claim 2.

Claim 7. *There exists a probabilistic machine $\mathcal{M}_{13}$, whose running time is at most that of $\mathcal{M}$, such that*

$$|\mathsf{Adv}_1^{(2)} - \mathsf{Adv}_2^{(2)}| \leq 2s(k) \cdot \max\{\mathsf{Adv}\mathrm{PRF}_{\tilde{\mathsf{F}},\mathcal{M}_{13}}(k), \mathsf{Adv}\mathrm{PRF}_{\hat{\mathsf{F}},\mathcal{M}_{13}}(k)\}.$$

Claim 8.

$$\mathsf{Adv}_2^{(2)} = \mathsf{Adv}_3^{(2)}$$

This is clear since the change is purely conceptual.

Claim 9. *There exists a probabilistic machine $\mathcal{M}_{14}$, whose running time is at most that of $\mathcal{M}$, such that*

$$|\mathsf{Adv}_3^{(2)} - \mathsf{Adv}_4^{(2)}| = \mathsf{Adv}\mathrm{DDH}_{\mathcal{M}_{14}}(k).$$

Proof. Given a DDH problem $\rho \leftarrow (\mathbb{G}, U, V, W, Z)$, where $\rho \xleftarrow{\mathsf{U}} \mathbb{D}(k)$ or $\rho \xleftarrow{\mathsf{U}} \mathbb{R}(k)$, we construct its adversary $\mathcal{M}_{14}$ using $\mathcal{M}_{11}$ in game $\mathbf{G}_3^{(2)}$ as follows:

$\mathcal{M}_{14}$ sets up the parameters of game $\mathbf{G}_3^{(2)}$ for two parties, $\mathcal{A}$ and $\mathcal{B}$, correctly and simulates the game with adversary $\mathcal{M}_{11}$ except the computation of g_1, g_2, X_1^*, X_2^*.

For the computation, $\mathcal{M}_{14}$ sets $g_1 \leftarrow U$, $g_2^* \leftarrow V$, $X_1^* \leftarrow W$, and $X_2^* \leftarrow Z$. (Note that the simulation of the other values can be perfectly executed with using g_1, g_2, X_1^*, X_2^*.)

Finally $\mathcal{M}_{14}$ outputs 1 iff $\mathcal{M}_{11}$ correctly guesses b^* (i.e., $\mathcal{M}_{11}$'s output b is equivalent to b^* in (Definition 2 of) game $\mathbf{G}_3^{(2)}$).

If $\rho \stackrel{\mathsf{U}}{\leftarrow} \mathbb{D}(k)$, the advantage of $\mathcal{M}_{11}$ in this simulation is equivalent to that in game $\mathbf{G}_3^{(2)}$, $\mathsf{Adv}_3^{(2)}$. Otherwise, the advantage of $\mathcal{M}_{11}$ in this simulation is equivalent to that in game $\mathbf{G}_4^{(2)}$, $\mathsf{Adv}_4^{(2)}$.

Therefore, $|\mathsf{Adv}_3^{(2)} - \mathsf{Adv}_4^{(2)}| = \mathsf{Adv}\mathrm{DDH}_{\mathcal{M}_{14}}(k)$. □

Claim 10. *There exists a probabilistic machine $\mathcal{M}_{15}$, whose running time is at most that of $\mathcal{M}$, such that*

$$|\mathsf{Adv}_4^{(2)} - \mathsf{Adv}_5^{(2)}| \leq s(k) \cdot \mathsf{Adv}\mathrm{TCR}_{\mathcal{M}_{15}}(k).$$

Proof. Given a TCR hash function problem $(\rho^*, h_B, \mathcal{D}_H, \mathcal{R}_H)$, where $h_B \stackrel{\mathsf{R}}{\leftarrow} \mathsf{KH}_k$, $\mathcal{D}_H \leftarrow \Pi_k \times \mathbb{G}^4$, $\mathcal{R}_H \leftarrow \mathbb{Z}_p$, $\rho^* \leftarrow (\mathsf{cert}_B, B_1, B_2, Y_1^*, Y_2^*) \stackrel{\mathsf{U}}{\leftarrow} \mathcal{D}_H$ and Π_k denotes the space of possible certificates, we construct its adversary $\mathcal{M}_{15}$ using $\mathcal{M}_{11}$ in game $\mathbf{G}_4^{(2)}$ as follows:

$\mathcal{M}_{15}$ simulates game $\mathbf{G}_4^{(2)}$ with adversary $\mathcal{M}_{11}$ with setting $(\mathsf{cert}_B, B_1, B_2)$ as the static public key of the peer (say $\mathcal{B}$) of the target session (say $\mathcal{A}$ for the owner), and setting (X_1^*, X_2^*, X_3^*) as the ephemeral public key of $\mathcal{A}$. Since the distributions of $(\mathsf{cert}_B, B_1, B_2)$ and (X_1^*, X_2^*) are equivalent to those of game $\mathbf{G}_4^{(2)}$ (e.g., the ephemeral public key of the target session, (X_1^*, X_2^*), is uniformly distributed on $\mathbb{G}^2$ in game $\mathbf{G}_4^{(2)}$). Therefore simulation of game $\mathbf{G}_4^{(2)}$ by $\mathcal{M}_{15}$ is perfect.

If $\mathcal{M}_{11}$ issues a session key query on session $(\hat{C}^{(i)}, \hat{B}, X_1^{(i)}, X_2^{(i)}, X_3^{(i)}, Y_1^{(i)}, Y_2^{(i)}, Y_3^{(i)})$ with $H_B(\hat{B}, X_1^*, X_2^*) = H_B(\hat{B}, X_1^{(i)}, X_2^{(i)})$ and $(\hat{B}, X_1^*, X_2^*) \neq (\hat{B}, X_1^{(i)}, X_2^{(i)})$ in this simulation, $\mathcal{M}_{15}$ outputs $(\hat{B}, X_1^{(i)}, X_2^{(i)})$.

Clearly, $\mathcal{M}_{15}$ outputs $(\hat{B}, X_1^{(i)}, X_2^{(i)})$ that breaks the TCR hash function, if game $\mathbf{G}_5^{(2)}$ applies the special rejection rule and aborts.

Since $\mathcal{M}_{15}$ has at most $s(k)$ sessions with $\mathcal{B}$, we obtain

$$|\mathsf{Adv}_4^{(2)} - \mathsf{Adv}_5^{(2)}| \leq s(k) \cdot \mathsf{Adv}\mathrm{TCR}_{\mathcal{M}_{15}}(k).$$

□

Claim 11. *There exists a probabilistic machine $\mathcal{M}_{16}$, whose running time is at most that of $\mathcal{M}$, such that*

$$|\ \mathsf{Adv}_6^{(2)} - \mathsf{Adv}_5^{(2)}\ | < \mathsf{Adv}\pi\mathrm{PRF}_{\mathsf{F}, I_{\mathbb{G}}, \mathcal{M}_{16}}(k) + 4/p.$$

Proof. Let $\mathsf{sid}_i \leftarrow (\hat{C}^{(i)}, \hat{B}, X_1^{(i)}, X_2^{(i)}, X_3^{(i)}, Y_1^{(i)}, Y_2^{(i)}, Y_3^{(i)})$ $(i = 1, \ldots, t(k))$ be sessions with $\mathcal{B}$ on which $\mathcal{M}_{11}$ issues session key queries, where $\mathcal{C}^{(i)}$ is a dishonest party established by $\mathcal{M}_{11}$. Let $K_i \leftarrow F_{\sigma_i}(\mathsf{sid}_i)$, where $\sigma_i \leftarrow (X_1^{(i)})^{b_1^*+d_i b_3^*}(X_2^{(i)})^{b_2^*+d_i b_4^*}$ $(X_3^{(i)})^{y_3^{(i)}}(C_1^{(i)})^{y^{(i)}}(C_2^{(i)})^{c_i y^{(i)}}$, $c_i \leftarrow H_C^{(i)}(\hat{C}^{(i)}, Y_1^{(i)}, Y_2^{(i)})$ and $d_i \leftarrow H_B(\hat{B}, X_1^{(i)}, X_2^{(i)})$.

Let the target session of game $\mathbf{G}_5^{(2)}$ be $\mathsf{sid}^* \leftarrow (\hat{A}, \hat{B}, X_1^*, X_2^*, X_3^*, Y_1^*, Y_2^*, Y_3^*)$ and the session key of sid^* be $K^* \leftarrow F_{\sigma^*}(\mathsf{sid}^*)$, where $\sigma^* \leftarrow (X_1^*)^{b_1^*+d^*b_3^*}(X_2^*)^{b_2^*+d^*b_4^*}$ $(Y_3^*)^{x_3^*}$ $(Y_1^*)^{a_1^*+c^*a_3^*}(Y_2^*)^{a_2^*+c^*a_4^*}$, $c^* \leftarrow H_A(\hat{A}, Y_1^*, Y_2^*)$ and $d^* \leftarrow H_B(\hat{B}, X_1^*, X_2^*)$.

We now consider two cases for each session sid_i $(i = 1, 2, \ldots, t(k))$, Case (i) and Case (ii).

Case (i): $(\mathbb{G}, g_1, g_2, X_1^{(i)}, X_2^{(i)}) \in \mathbb{D}(k)$. That is, there exists $x \in \mathbb{Z}_p$ such that $X_1^{(i)} = g_1^x, X_2^{(i)} = g_2^x$.

Case (ii): $(\mathbb{G}, g_1, g_2, X_1^{(i)}, X_2^{(i)}) \notin \mathbb{D}(k)$.

We say $(\mathbb{G}, g_1, g_2, X_1^*, X_2^*) \in \mathsf{GoodKey}$, if $(\mathbb{G}, g_1, g_2, X_1^*, X_2^*) \notin \mathbb{D}(k)$ and $g_1 \neq 1, g_2 \neq 1, g_1 \neq g_2$. Since the parameter is uniformly selected from $\mathbb{R}(k)$ in game $\mathbf{G}_5^{(2)}$, it occurs with probability at least $(1-4/p)$. Hereafter, we assume that $(\mathbb{G}, g_1, g_2, X_1^*, X_2^*) \in \mathsf{GoodKey}$ occurs in game $\mathbf{G}_5^{(2)}$. Note that all games to be investigated here are modified from game $\mathbf{G}_5^{(2)}$ with preserving the distribution of $(\mathbb{G}, g_1, g_2, X_1^*, X_2^*)$.

$(B_1, B_2, \sigma^*, \sigma_i)$ are expressed by the following equations:

$$\begin{aligned}
\log_{g_1} B_1 &\equiv b_1^* + \eta b_2^* \pmod p\\
\log_{g_1} B_2 &\equiv b_3^* + \eta b_4^* \pmod p\\
\log_{g_1} \sigma^* &\equiv x_1^*(b_1^* + d^*b_3^*) + \eta x_2^*(b_2^* + d^*b_4^*) + \delta \pmod p\\
\log_{g_1} \sigma_i &\equiv x(b_1^* + d_i b_3^*) + \eta x(b_2^* + d_i b_4^*) + \gamma \pmod p.
\end{aligned}$$

where $g_2 = g_1^\eta$, $X_1^* = g_1^{x_1^*}$, $X_2^* = g_1^{x_2^*}$, $(Y_3^*)^{x_3^*}(Y_1^*)^{a_1^*+c^*a_3^*}(Y_2^*)^{a_2^*+c^*a_4^*} = g_1^\delta$ $X_1^{(i)} = g_1^x$, $X_2^{(i)} = g_1^x$ and $(X_3^{(i)})^{y_3^{(i)}}(C_1^{(i)})^{y^{(i)}}(C_2^{(i)})^{c_i y^{(i)}} n = g_1^\gamma$.

If Case (i) occurs, the value of σ_i is (information theoretically) independent from σ^* for any $i = 1, \ldots, t(k)$, since

$$\log_{g_1} \sigma_i - \gamma \equiv x(b_1^* + \eta b_2^*) + x d_i(b_3^* + \eta b_4^*) \pmod p$$

is linearly dependent to $\log_{g_1} B_1$ and $\log_{g_1} B_2$, while $\log_{g_1} \sigma^*$ is linearly independent from $\log_{g_1} B_1$ and $\log_{g_1} B_2$. (Actually, given sid_i, the value of σ_i is uniquely determined, but, given sid^*, the value of σ^* is still uniformly distributed in $\mathbb{G}$ if $(b_2^*, b_4^*) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$.)

Hence, hereafter we consider the case that Case (ii) occurs for all $i = 1, \ldots, t(k)$. Then, we will show the following proposition:

Proposition 1. *Let assume that* $(\mathbb{G}, g_1, g_2, X_1^*, X_2^*) \in \mathsf{GoodKey}$. *Then, given* $(\mathsf{sid}^*, \mathsf{sid}_1, \ldots, \mathsf{sid}_{t(k)})$, σ^* *and* σ_i *are pairwisely independent for any* $i = 1, \ldots, t(k)$, *and each one is uniformly distributed over* $\mathbb{G}$, *when* $(b_2^*, b_4^*) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$.

Proof. First, we consider the relation between sid^* and sid_i $(i = 1, \ldots, t(k))$. So we investigate the following matrix:

$$\begin{bmatrix} 1 & \eta & 0 & 0 \\ 0 & 0 & 1 & \eta \\ x_1^* & \eta x_2^* & d^* x_1^* & \eta d^* x_2^* \\ x_1 & \eta x_2 & d_i x_1 & \eta d_i x_2 \end{bmatrix}, \tag{1}$$

where $X_1^{(i)} = g_1^{x_1}$ and $X_2^{(i)} = g_1^{x_2}$.

This matrix (1) is regular if and only if

$$\eta^2(x_2^* - x_1^*)(x_2 - x_1)(d^* - d_i) \not\equiv 0 \pmod{p}. \tag{2}$$

$\eta \neq 0$ and $x_2^* - x_1^* \neq 0$, since we assume that $(\mathbb{G}, g_1, g_2, X_1^*, X_2^*) \in \mathsf{GoodKey}$, i.e., $(\mathbb{G}, g_1, g_2, X_1^*, X_2^*) \notin \mathbb{D}(k)$ and $g_1 \neq 1, g_2 \neq 1, g_1 \neq g_2$. In game $\mathbf{G}_5^{(2)}$, $d^* - d_i \neq 0$ by the special rejection rule, and $x_2 - x_1 \neq 0$ in Case (ii).

Therefore, this matrix (1) is regular. Then, given (sid^*,sid_i), the value of (σ_i, σ^*) is uniformly distributed over $\mathbb{G}^2$ when $(b_2^*, b_4^*) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$. □

We can then construct an adversary $\mathcal{M}_{16}$ for πPRF F with index $\{(I_\mathbb{G}, f_\mathbb{G})\}_{\mathbb{G}\in\{\mathbb{G}\}_k, k\in\mathbb{N}}$, by using $\mathcal{M}_{11}$ in game $\mathbf{G}_5^{(2)}$ as follows, where $I_\mathbb{G} \leftarrow \{(V, W, d) \mid (V, W, d) \in \mathbb{G}^2 \times \mathbb{Z}_p\}$ and $f_\mathbb{G} : (V, W, d) \mapsto V^{r_1 + d r_2} W$ with $(r_1, r_2) \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$:

Given the πPRF security experiment for F, $\mathcal{M}_{16}$ sets up the parameters of game $\mathbf{G}_5^{(2)}$ such that

$$\begin{aligned}
&\mathbb{G} \stackrel{\mathsf{U}}{\leftarrow} \{\mathbb{G}\}_k, g_1 \stackrel{\mathsf{U}}{\leftarrow} \mathbb{G}, \quad \eta \stackrel{\mathsf{U}}{\leftarrow} \mathbb{Z}_p, \quad g_2 \leftarrow g_1^{\eta}, \\
&(a_1^*, a_2^*, a_3^*, a_4^*) \stackrel{\mathsf{U}}{\leftarrow} (\mathbb{Z}_p)^4, \quad A_1 \leftarrow g_1^{a_1^*} g_2^{a_2^*}, A_2 \leftarrow g_1^{a_3^*} g_2^{a_4^*}, \\
&(\beta_1, \beta_2) \stackrel{\mathsf{U}}{\leftarrow} (\mathbb{Z}_p)^2, \quad B_1 \leftarrow g_1^{\beta_1}, \quad B_2 \leftarrow g_1^{\beta_2}, \\
&(x_1^*, x_2^*, x_3^*) \stackrel{\mathsf{U}}{\leftarrow} (\mathbb{Z}_p)^2 (x_1^* \neq x_2^*), \quad X_1^* \leftarrow g_1^{x_1^*}, \quad X_2^* \leftarrow g_2^{x_2^*}, \quad X_3^* \leftarrow g_2^{x_3^*} \\
&(y^{(i)}, y_3^{(i)}) \stackrel{\mathsf{U}}{\leftarrow} (\mathbb{Z}_p), \quad Y_1^{(i)} = g_1^{y^{(i)}}, \quad Y_2^{(i)} = g_2^{y^{(i)}}, Y_3^{(i)} = g_1^{y_3^{(i)}} \\
&c^* \leftarrow H_A(\hat{A}, Y_1^*, Y_2^*), \quad d^* \leftarrow H_B(\hat{B}, X_1^*, X_2^*) \\
&c_i \leftarrow H_C(\hat{C}^{(i)}, Y_1^{(i)}, Y_2^{(i)}), \quad d_i \leftarrow H_B(\hat{B}, X_1^{(i)}, X_2^{(i)}), \\
&V^* \leftarrow X_2^*/(X_1^*)^\eta, \quad W^* \leftarrow (X_1^*)^{\beta_1 + d^*\beta_2} (Y_3^*)^{x_3^*} (Y_1^*)^{a_1^* + c^* a_3^*} (Y_2^*)^{a_2^* + c^* a_4^*}, \\
&V_i \leftarrow X_2^{(i)}/(X_1^{(i)})^\eta, \quad W_i \leftarrow (X_1^{(i)})^{\beta_1 + d_i \beta_2} (X_3^{(i)})^{y_3^{(i)}} (C_1^{(i)})^{y^{(i)}} (C_2^{(i)})^{c_i y^{(i)}} \\
&(i = 1, \ldots, t(k)).
\end{aligned}$$

Under this setting of the parameters, $\mathcal{M}_{16}$ can perfectly simulate the sessions, sid^* and sid_i, with $\mathcal{M}_{11}$ except the computation of σ^* and σ_i, for $i = 1, \ldots, t(k)$.

We now set $(r_1, r_2) \leftarrow (b_2^*, b_4^*)$, and apply the index, $I_\mathbb{G} \leftarrow \{(V, W, d) \mid (V, W, d) \in \mathbb{G}^2 \times \mathbb{Z}_p\}$ and $f_\mathbb{G} : (V, W, d) \mapsto V^{r_1 + d r_2} W$. Then, $\sigma_{(V^*, W^*, d^*)} = \sigma^*$ and $\sigma_{(V_i, W_i, d_i)} =$

σ_i for $i = 1, \ldots, t(k)$, because

$$\begin{aligned}
&\sigma_{(V^*,W^*,d^*)} = (V^*)^{r_1+d^*r_2} W^* \\
&= (X_2^*)^{b_2^*}/(X_1^*)^{\eta b_2^*} \cdot (X_2^*)^{d^* b_4^*}/(X_1^*)^{\eta d^* b_4^*} \cdot (X_1^*)^{\beta_1+d^*\beta_2} \cdot (Y_3^*)^{x_3^*}(Y_1^*)^{a_1^*+c^*a_3^*}(Y_2^*)^{a_2^*+c^*a_4^*} \\
&= (X_1^*)^{b_1^*+d^*b_3^*}(X_2^*)^{b_2^*+d^*b_4^*} \cdot (Y_3^*)^{x_3^*}(Y_1^*)^{a_1^*+c^*a_3^*}(Y_2^*)^{a_2^*+c^*a_4^*} \\
&= \sigma^*, \\
&\sigma_{(V_i,W_i,d_i)} = V_i^{r_1+d_ir_2} W_i \\
&= (X_2^{(i)})^{b_2^*}/(X_1^{(i)})^{\eta b_2^*} \cdot (X_2^{(i)})^{d_i b_4^*}/(X_1^{(i)})^{\eta d_i b_4^*} \cdot (X_1^{(i)})^{\beta_1+d_i\beta_2}(X_3^{(i)})^{y_3^{(i)}}(C_1^{(i)})^{y^{(i)}}(C_2^{(i)})^{c_i y^{(i)}} \\
&= (X_1^{(i)})^{b_1^*+d_ib_3^*}(X_2^{(i)})^{b_2^*+d_ib_4^*} \cdot (X_3^{(i)})^{y_3^{(i)}}(C_1^{(i)})^{y^{(i)}}(C_2^{(i)})^{c_i y^{(i)}}, \\
&= \sigma_i,
\end{aligned}$$

where $b_1^* \equiv \beta_1 - \eta b_2^* \pmod{p}$, and $b_3^* \equiv \beta_2 - \eta b_4^* \pmod{p}$. Here note that $\sigma_{(V^*,W^*,d^*)} = \sigma^*$ and $\sigma_{(V_i,W_i,d_i)} = \sigma_i$ for $i = 1, \ldots, t(k)$ hold simultaneously for any selected value of (b_2^*, b_4^*).

$\mathcal{M}_{16}$ executes game $\mathbf{G}_5^{(2)}$ except the computation of $F_{\sigma^*}(\mathsf{sid}^*)$ and $F_{\sigma_i}(\mathsf{sid}_i)$ ($i = 1, \ldots, t(k)$), and $\mathcal{M}_{16}$ gives index (V^*, W^*, d^*) and (V_i, W_i, d_i) ($i = 1, \ldots, t(k)$) to the oracle in the experiment of the πPRF security definition (in Section 2.4). If the oracle is for $\mathcal{A}^{F,I_{\mathbb{G}}}$, the above-mentioned simulation is the same as game $\mathbf{G}_5^{(2)}$. If the oracle is for $\mathcal{A}^{RF,I_\Sigma}$, the simulation is the same as game $\mathbf{G}_6^{(2)}$.

From Proposition 1, if $\mathsf{GoodCoin}$ occurs, (σ^*, σ_i) are pairwisely independent for $i = 1, \ldots, t(k)$.

Since $\Pr[\neg\mathsf{GoodCoin}] < 4/p$, we then obtain

$$| \mathsf{Adv}_6^{(2)} - \mathsf{Adv}_5^{(2)} | < \mathsf{Adv}\pi\mathrm{PRF}_{\mathsf{F},I_{\mathbb{G}},\mathcal{M}_{16}}(k) + 4/p. \tag{3}$$

□

Since $\mathsf{Adv}_6^{(2)} = 0$, by summing up Claims 6 to 11, we obtain the following claim,

Claim 12. *Let's suppose Case 2 occurs. For any adversary $\mathcal{M}$ in the eCK game (Definition 2), there exist probabilistic machines, $\mathcal{M}_{12}, \mathcal{M}_{13}, \mathcal{M}_{14}, \mathcal{M}_{15}$ and $\mathcal{M}_{16}$, whose running times are at most that of $\mathcal{M}$, such that*

$$\begin{aligned}
&\mathsf{Adv}\mathrm{AKE}_{\mathcal{M}}(k) < 2n(k)^2 s(k) \cdot (2s(k) \cdot \max\{\mathsf{Adv}\mathrm{PRF}_{\tilde{\mathsf{F}},\mathcal{M}_{13}}(k), \mathsf{Adv}\mathrm{PRF}_{\hat{\mathsf{F}},\mathcal{M}_{13}}(k)\} + \\
&\quad \mathsf{Adv}\mathrm{DDH}_{\mathcal{M}_{14}}(k) + s(k) \cdot \mathsf{Adv}\mathrm{TCR}_{\mathcal{M}_{15}}(k) + \mathsf{Adv}\pi\mathrm{PRF}_{\mathsf{F},I_{\mathbb{G}},\mathcal{M}_{16}}(k) + 4/p) + \\
&\quad s(k) \cdot \mathsf{Adv}\mathrm{PRF}_{\tilde{\mathsf{F}},\mathcal{M}_{12}}(k).
\end{aligned}$$

□

4 The Proposed KEM Scheme

4.1 Scheme

In this section, we present a CCA secure KEM scheme.

Let $k \in \mathbb{N}$ be a security parameter, and let $\mathbb{G} \overset{\mathsf{U}}{\leftarrow} \{\mathbb{G}\}_k$ be a group with security parameter k, where the order of $\mathbb{G}$ is prime p and $|p| = k$.

Let H be a TCR hash function family, and F be a πPRF family with index $\{(I_{\mathbb{G}}, f_{\mathbb{G}})\}_{\mathbb{G} \in \{\mathbb{G}\}_k, k \in \mathbb{N}}$, where $I_{\mathbb{G}} \leftarrow \{(V, W, d) \mid (V, W, d) \in \mathbb{G}^2 \times \mathbb{Z}_p\}$ and $f_{\mathbb{G}} : (V, W, d) \mapsto V^{r_1 + d r_2} W$ with $(r_1, r_2) \overset{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$.

Secret Key: The secret key is $sk \leftarrow (x_1, x_2, y_1, y_2) \overset{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^4$.

Public Key: $g_1 \overset{\mathsf{U}}{\leftarrow} \mathbb{G}$, $g_2 \overset{\mathsf{U}}{\leftarrow} \mathbb{G}$, $z \leftarrow g_1^{x_1} g_2^{x_2}$, $w \leftarrow g_1^{y_1} g_2^{y_2}$, $H \leftarrow \mathsf{H}_h^{k, \mathbb{G}^4, \mathbb{Z}_p}$ and $F \leftarrow \mathsf{F}^{k, \Sigma_{\mathsf{F}}, \mathcal{D}_{\mathsf{F}}, \mathcal{R}_{\mathsf{F}}}$, where $h \overset{\mathsf{R}}{\leftarrow} \mathsf{KH}_k$, $\Sigma_{\mathsf{F}} \leftarrow \mathbb{G}$, $\mathcal{D}_{\mathsf{F}} \leftarrow \{pk\} \times \mathbb{G}^2$ (pk is a possible public-key value) and $\mathcal{R}_{\mathsf{F}} \leftarrow \{0, 1\}^k$.

The public key is $pk \leftarrow (\mathbb{G}, g_1, g_2, z, w, H, F)$.

Encryption: Choose $r \overset{\mathsf{U}}{\leftarrow} \mathbb{Z}_p$ and compute

$$
\begin{aligned}
&C_1 \leftarrow g_1^r, \\
&C_2 \leftarrow g_2^r, \\
&d \leftarrow H(z, w, C_1, C_2) \\
&\sigma \leftarrow z^r w^{rd} \\
&K \leftarrow F_\sigma(pk, C_1, C_2).
\end{aligned}
$$

(C_1, C_2) is a ciphertext, and K is the secret key to be shared.

Decryption: Given (z, w, C_1, C_2), check whether

$$(z, w, C_1, C_2) \in \mathbb{G}^4.$$

If it holds, compute

$$
\begin{aligned}
&d \leftarrow H(z, w, C_1, C_2) \\
&\sigma \leftarrow C_1^{x_1 + d y_1} C_2^{x_2 + d y_2} \\
&K \leftarrow F_\sigma(pk, C_1, C_2).
\end{aligned}
$$

4.2 CCA Security

Theorem 2. *The proposed KEM scheme is IND-CCA2 secure, if the DDH assumption holds for $\{\mathbb{G}\}_{k \in \mathbb{N}}$, H is a TCR hash function family, and F is a πPRF family with index $\{(I_{\mathbb{G}}, f_{\mathbb{G}})\}_{\mathbb{G} \in \{\mathbb{G}\}_k, k \in \mathbb{N}}$, where $I_{\mathbb{G}} \leftarrow \{(V, W, d) \mid (V, W, d) \in \mathbb{G}^2 \times \mathbb{Z}_p\}$ and $f_{\mathbb{G}} : (V, W, d) \mapsto V^{r_1 + d r_2} W$ with $(r_1, r_2) \overset{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$.*

Proof. The proof is similar to the proof of the security of the proposed AKE in Case 2.

First let us review some notation of the IND-CCA2 game of our scheme. Let (C_1^*, C_2^*), $pk^* \leftarrow (\mathbb{G}, g_1, g_2, z^*, w^*, H, F)$ and $(x_1^*, x_2^*, y_1^*, y_2^*)$ be the target ciphertext, public key and secret key, and $K^* \leftarrow F_{\sigma^*}(pk^*, C_1^*, C_2^*)$, where $d^* \leftarrow H(z^*, w^*, C_1^*, C_2^*)$ and $\sigma^* \leftarrow (z^*)^{r^*} {w^*}^{r^* d^*}$. When adversary $\mathcal{A}$ sends $(C_1^{(i)}, C_2^{(i)})$ to the decryption oracle DO ($i = 1, \ldots, t(k)$), the oracle returns $K \leftarrow F_\sigma(pk^*, C_1^{(i)}, C_2^{(i)})$, where $d \leftarrow H(z^*, w^*, C_1^{(i)}, C_2^{(i)})$ and $\sigma \leftarrow (C_1^{(i)})^{x_1^*+dy_1^*}(C_2^{(i)})^{x_2^*+dy_2^*}$.

In this proof, we consider five games, $\mathbf{G}_0$, $\mathbf{G}_1$, $\mathbf{G}_2$, $\mathbf{G}_3$ and $\mathbf{G}_4$ as follows:

Game $\mathbf{G}_0$. This is the original IND-CCA2 game with adversary $\mathcal{A}$ to define $\mathsf{Adv}\mathrm{KEM}_{\mathcal{A}}^{\text{IND-CCA2}}(k)$.

Game $\mathbf{G}_1$. We modify game $\mathbf{G}_0$ to game $\mathbf{G}_1$ by changing $\sigma^* \leftarrow (z^*)^{r^*} w^{*r^*d^*}$ (in the computation process of the target ciphertext K^* in the encryption oracle) to $\sigma^* \leftarrow C_1^{*x_1^*+d^*y_1^*}(C_2^*)^{x_2^*+d^*y_2^*}$. This change is purely conceptual.

Game $\mathbf{G}_2$. We modify game $\mathbf{G}_1$ to game $\mathbf{G}_2$ by changing DH tuple $(\mathbb{G}, g_1, g_2, C_1^*, C_2^*) \overset{\mathsf{U}}{\leftarrow} \mathbb{D}(k)$ to a random tuple $(\mathbb{G}, g_1, g_2, C_1^*, C_2^*) \overset{\mathsf{U}}{\leftarrow} \mathbb{R}(k)$.

Game $\mathbf{G}_3$. We modify game $\mathbf{G}_2$ to game $\mathbf{G}_3$ by adding a special rejection rule to game $\mathbf{G}_2$, such that, game $\mathbf{G}_3$ aborts if $\mathcal{A}$ sends $(C_1^{(i)}, C_2^{(i)})$ to the decryption oracle and $H(z^*, w^*, C_1^*, C_2^*) = H(z^*, w^*, C_1^{(i)}, C_2^{(i)})$ and $(z^*, w^*, C_1^*, C_2^*) \neq (z^*, w^*, C_1^{(i)}, C_2^{(i)})$ occur.

Game $\mathbf{G}_4$. We modify game $\mathbf{G}_3$ to game $\mathbf{G}_4$ by changing πPRF F_{σ^*} in the the encryption oracle to a random function.

Let Adv_0 be the IND-CCA2 advantage of $\mathcal{A}$ in game $\mathbf{G}_0$ (i.e., $\mathsf{Adv}\mathrm{KEM}_{\mathcal{A}}^{\text{IND-CCA2}}(k)$). Let Adv_i ($i = 1, 2, 3, 4$) be the IND-CCA2 advantage of $\mathcal{A}$ in game $\mathbf{G}_i$.

We can obtain the following claims, whose proofs are essentially the same as those of Claims 8, 9, 10 and 11. So we omit them here.

Claim 13. $\mathsf{Adv}_0 = \mathsf{Adv}_1$

Claim 14. *There exists a probabilistic machine $\mathcal{A}_1$, whose running time is at most that of $\mathcal{A}$, such that* $|\mathsf{Adv}_1 - \mathsf{Adv}_2| = \mathsf{Adv}\mathrm{DDH}_{\mathcal{A}_1}(k)$.

Claim 15. *There exists a probabilistic machine $\mathcal{A}_2$, whose running time is at most that of $\mathcal{A}$, such that* $|\mathsf{Adv}_2 - \mathsf{Adv}_3| \leq t(k) \cdot \mathsf{Adv}\mathrm{TCR}_{\mathcal{A}_2}(k)$.

Claim 16. *There exists a probabilistic machine $\mathcal{A}_3$, whose running time is at most that of $\mathcal{A}$, such that* $|\mathsf{Adv}_3 - \mathsf{Adv}_4| < \mathsf{Adv}\pi\mathrm{PRF}_{\mathsf{F}, I_{\mathbb{G}}, \mathcal{A}_3}(k) + 4/p$.

Since $\mathsf{Adv}_4 = 0$, by summing up Claims 13 to 16, we obtain the following claim,

Claim 17. *For any adversary $\mathcal{A}$ in the IND-CCA2 game there exist probabilistic machines, $\mathcal{A}_1, \mathcal{A}_2$ and $\mathcal{A}_3$ whose running times are at most that of $\mathcal{A}$, such that*

$$\mathsf{Adv}\mathrm{KEM}_{\mathcal{A}}^{\textit{IND-CCA2}}(k) \leq \mathsf{Adv}\mathrm{DDH}_{\mathcal{A}_1}(k) + t(k) \cdot \mathsf{Adv}\mathrm{TCR}_{\mathcal{A}_2}(k) + \mathsf{Adv}\pi\mathrm{PRF}_{\mathsf{F}, I_{\mathbb{G}}, \mathcal{A}_3}(k) + 4/p.$$

□

4.3 CPCA Security

In this paper, we define a stronger security notion than the CCA security on KEM and PKE.

Here, we consider a trapdoor commitment, where committer (sender) $\mathcal{S}$ commits to x by sending $C \leftarrow \mathsf{E}_{pk}(x)$ to receiver $\mathcal{R}$, then $\mathcal{S}$ opens x by sending sk to $\mathcal{R}$, where

(pk, sk) is a pair of public key and secret key, and $x = \mathsf{D}_{sk}(C)$. Using a trapdoor commitment, several committers, $\mathcal{S}_1$, ..., $\mathcal{S}_n$, commits to $x_1, \ldots, x_n$ respectively by sending $C_1 \leftarrow \mathsf{E}_{pk}(x_1)$, ..., $C_n \leftarrow \mathsf{E}_{pk}(x_n)$ to receiver $\mathcal{R}$. Another party can open them simultaneously by sending sk to receiver $\mathcal{R}$. A possible malleable attack is as follows: after looking at pk and $C \leftarrow \mathsf{E}_{pk}(x)$ sent to receiver $\mathcal{R}$, adversary $\mathcal{A}$ computes pk', C', algorithm Conv and non-trivial relation Rel. $\mathcal{A}$ registers pk' and sends C' to $\mathcal{R}$ as a commitment to x' such that $\mathsf{Rel}(x, x')$. When sk is opened, $\mathcal{A}$ computes $sk' \leftarrow \mathsf{Conv}(sk)$ and sends sk' to $\mathcal{R}$ such that $x' = \mathsf{D}_{sk'}(C')$.

To capture the security against such malleable attacks, we now define the CPCA (Chosen Public-key and Ciphertext Attacks) security for KEM schemes.

Definition 3. *Let $\Sigma = (\mathsf{K}, \mathsf{E}, \mathsf{D})$ be a KEM scheme. Let C^*, pk^* and sk^* be the target ciphertext, public key and secret key of KEM scheme Σ. In the CPCA security, an adversary $\mathcal{A}$, given pk^* and C^*, is allowed to submit a ciphertext C along with polynomial-time algorithm, $\mathsf{Conv} \leftarrow (\mathsf{Conv}_1, \mathsf{Conv}_2)$, to the decryption oracle DO (with sk^*) under the condition that $(\mathsf{Conv}_1(pk^*), C) \neq (pk^*, C^*)$, where Conv_1 and Conv_2 uniquely compute the public-key $pk \leftarrow \mathsf{Conv}_1(pk^*)$ and the corresponding secret key $sk \leftarrow \mathsf{Conv}_2(sk^*, pk^*)$, respectively. Here there exists a polynomial-time algorithm of verifying the validity of Conv such that for all $(c, k) \leftarrow \mathsf{E}_{\mathsf{Conv}_1(pk^*)}(1^k)$ $k = \mathsf{D}_{\mathsf{Conv}_2(sk^*,pk^*)}(c)$. If Conv is valid, DO computes $sk \leftarrow \mathsf{Conv}_2(sk^*, pk^*)$ and returns $\mathsf{D}_{sk}(C)$ to $\mathcal{A}$.*

We can define the advantage of $\mathcal{A}$ for the IND-CPCA game, $\mathsf{Adv}\mathrm{KEM}_{\mathcal{A}}^{\textit{IND-CPCA}}(k)$. We say that a KEM scheme is IND-CPCA-secure if for any probabilistic polynomial-time (PPT) adversary $\mathcal{A}$, $\mathsf{Adv}\mathrm{KEM}_{\mathcal{A}}^{\textit{IND-CPCA}}(k)$ is negligible in k.

This notion is considered to be closely related to the notion, *complete non-malleability*, introduced by Fischlin [4].

We now show that the proposed KEM scheme is CPCA secure. To prove the security, we need a new requirement for a hash function family, the generalized TCR (GTCR) hash function family.

Definition 4. *Let $k \in \mathbb{N}$ be a security parameter. Let H be a hash function family associated with Dom_k, Rng_k and KH_k.*

Let Conv and Rel be function and relation families with parameter space Param_k. Let $\tau \in \mathsf{Param}_k$, then function $\mathsf{Conv}_\tau : X_k \to X_k$ maps $e_1 \in X_k$ to $e_2 \in X_k$. Given $\mathcal{R} \overset{\mathsf{R}}{\leftarrow} \mathsf{Rng}_k$ of hash function family H, $\mathsf{Rel}_\tau \subset \mathcal{R} \times \mathcal{R}$ is an associated relation of H, where, for any $d_1 \in \mathcal{R}$, $d_2 \in \mathcal{R}$ is uniquely determined with $\mathsf{Rel}_\tau(d_1, d_2)$.

Let $\mathcal{A}$ be a probabilistic polynomial-time machine. For all k, we define

$$\mathsf{Adv}\mathrm{GTCR}_{\mathsf{H},\mathcal{A}}^{\mathsf{Conv},\mathsf{Rel}}(k) \leftarrow \Pr[\ \mathsf{Rel}_\tau(\mathsf{H}_h^{k,\mathcal{D},\mathcal{R}}(\rho, \delta), \mathsf{H}_h^{k,\mathcal{D},\mathcal{R}}(\mathsf{Conv}_\tau(\rho), \delta'))\ \mid$$
$$(\tau, \delta') \overset{\mathsf{R}}{\leftarrow} \mathcal{A}(1^k, \rho, \delta, h)\ \wedge\ (\rho, \delta) \neq (\mathsf{Conv}_\tau(\rho), \delta')\ \wedge\ (\rho, \delta) \overset{\mathsf{U}}{\leftarrow} \mathcal{D}\ \wedge\ h \overset{\mathsf{R}}{\leftarrow} \mathsf{KH}_k].$$

Hash function family H is a generalized target collision resistant (GTCR) hash function family associated with $(\mathsf{Conv}, \mathsf{Rel})$ if for any probabilistic polynomial-time adversary $\mathcal{A}$, $\mathsf{Adv}\mathrm{GTCR}_{\mathsf{H},\mathcal{A}}^{\mathsf{Conv},\mathsf{Rel}}(k)$ is negligible in k.

If Conv_τ is a constant function to τ and $\mathsf{Rel}_\tau(d_1, d_1) \Leftrightarrow d_1 = d_2$, then the GTCR hash function family associated with $(\mathsf{Conv}, \mathsf{Rel})$ is a TCR hash function family.

Theorem 3. *The proposed KEM scheme is IND-CPCA secure, if the DDH assumption holds for* $\{\mathbb{G}\}_{k\in\mathbb{N}}$, H *is a GTCR hash function family associated with* $(\mathsf{Conv}, \mathsf{Rel})$*, and* F *is a* π*PRF family with index* $\{(I_{\mathbb{G}}, f_{\mathbb{G}})\}_{\mathbb{G}\in\{\mathbb{G}\}_k, k\in\mathbb{N}}$*, where*

– $(z, w) \leftarrow \mathsf{Conv}_{(u_1,u_2,v_1,v_2)}(z^*, w^*) \in \mathbb{G}^2$ *is defined by* $z \leftarrow (z^*)^{u_1}(w^*)^{u_2}$ *and* $w \leftarrow (z^*)^{v_1}(w^*)^{v_2}$*, and* $\mathsf{Rel}_{(u_1,u_2,v_1,v_2)}(d_1, d_2) \Leftrightarrow d_2(d_1v_1-v_2)+(d_1u_1-u_2) \equiv 0 \pmod p$*, where* $(u_1, u_2, v_1, v_2) \in \mathbb{Z}_p^4$*, and*
– $I_{\mathbb{G}} \leftarrow \{(V, W, d) \mid (V, W, d) \in \mathbb{G}^2 \times \mathbb{Z}_p\}$ *and* $f_{\mathbb{G}} : (V, W, d) \mapsto V^{r_1+dr_2}W$ *with* $(r_1, r_2) \overset{\mathsf{U}}{\leftarrow} \mathbb{Z}_p^2$*.*

Proof. We define five games, $\mathbf{G}_0$, $\mathbf{G}_1$, $\mathbf{G}_2$, $\mathbf{G}'_3$ and $\mathbf{G}'_4$, that are equivalent to the games defined in the proof of Theorem 2 except game $\mathbf{G}'_3$ and game $\mathbf{G}'_4$.

Game $\mathbf{G}'_3$. We modify game $\mathbf{G}_2$ to game $\mathbf{G}'_3$ by adding a special rejection rule to game $\mathbf{G}_2$, such that, game $\mathbf{G}'_3$ aborts if $\mathcal{A}$ sends $((u_1^{(i)}, u_2^{(i)}, v_1^{(i)}, v_2^{(i)}), (C_1^{(i)}, C_2^{(i)})) \in \mathbb{Z}_p^4 \times \mathbb{G}^2$ to the decryption oracle, the relation, $d_i(d^*v_1^{(i)} - v_2^{(i)}) + (d^*u_1^{(i)} - u_2^{(i)}) \equiv 0 \pmod p$, holds for $d^* \leftarrow H(z^*, w^*, C_1^*, C_2^*)$ and $d_i \leftarrow H(z_i, w_i, C_1^{(i)}, C_2^{(i)})$, and $(z^*, w^*, C_1^*, C_2^*) \neq (z_i, w_i, C_1^{(i)}, C_2^{(i)})$, where $z_i \leftarrow (z^*)^{u_1^{(i)}}(w^*)^{u_2^{(i)}}$ and $w_i \leftarrow (z^*)^{v_1^{(i)}}(w^*)^{v_2^{(i)}}$, for $i = 1, \ldots, t(k)$.

The difference of game $\mathbf{G}'_3$ and game $\mathbf{G}'_4$ is the same as that of game $\mathbf{G}_3$ and game $\mathbf{G}_4$.

Let Adv'_0 be the IND-CPCA advantage of $\mathcal{A}$ in game $\mathbf{G}_0$ (i.e., $\mathsf{Adv}\mathrm{KEM}_{\mathcal{A}}^{\text{IND-CPCA}}(k)$). Let Adv'_i $(i = 1, 2, 3, 4)$ be the IND-CPCA advantage of $\mathcal{A}$ in game $\mathbf{G}_i$ $(i = 1, 2)$ and $\mathbf{G}'_i$ $(i = 3, 4)$.

Claims 13 and 14 hold for this proof, and the following claim can be proven in a manner similar to Claim 15 (Claim 10).

Claim 18. $|\mathsf{Adv}'_2 - \mathsf{Adv}'_3| \leq t(k) \cdot \mathsf{Adv}\mathrm{GTCR}_{\mathsf{H},\mathcal{A}'_2}^{\mathsf{Conv},\mathsf{Rel}}(k)$.

In a manner similar to Claim 16 (Claim 11), we can show the following claim:
Claim 19. *There exists a probabilistic machine* $\mathcal{A}'_3$*, whose running time is at most that of* $\mathcal{A}$*, such that* $|\mathsf{Adv}'_3 - \mathsf{Adv}'_4| < \mathsf{Adv}\pi\mathrm{PRF}_{\mathsf{F},I_{\mathbb{G}},\mathcal{A}_3}(k) + 4/p$*.*

Proof. For any $((u_1^{(i)}, u_2^{(i)}, v_1^{(i)}, v_2^{(i)}), (C_1^{(i)}, C_2^{(i)}))$ queried to the decryption oracle, if $\log_{g_1} C_1^{(i)} \equiv \log_{g_2} C_2^{(i)} \pmod p$ (i.e, $(\mathbb{G}, g_1, g_2, C_1, C_2) \in \mathbb{D}(k)$), then it is the same as Case (i) in Claim 11.

So, we now only consider Case (ii) in Claim 11, $\log_{g_1} C_1^{(i)} \not\equiv \log_{g_2} C_2^{(i)} \pmod p$.

Since the values of (x_1^*, x_2^*) and (y_1^*, y_2^*) are information theoretically undetermined, only way for $\mathcal{A}$ to specify Conv to generate the secret key, (x_1, x_2, y_1, y_2), of (z, w) from $(x_1^*, x_2^*, y_1^*, y_2^*)$ is to use a linear relation over $\log_{g_1}$ of $\mathbb{G}$. That is, the most general form of the conversion of (z, w) from (z^*, w^*) is $z \leftarrow (z^*)^{u_1}(w^*)^{u_2}g_1^{s_1}g_2^{s_2}$ and $w \leftarrow (z^*)^{v_1}(w^*)^{v_2}g_1^{t_1}g_2^{t_2}$, and $(x_1, x_2) \leftarrow (u_1x_1^* + u_2y_1^* + s_1, u_1x_2^* + u_2y_2^* + s_2)$ and $(y_1, y_2) \leftarrow (v_1x_1^* + v_2y_1^* + t_1, v_1x_2^* + v_2y_2^* + t_2)$, where $(u_1, u_2, v_1, v_2, s_1, s_2, t_1, t_2) \in \mathbb{Z}_p^8$.

In our security analysis, the part of the conversion regarding (s_1, s_2, t_1, t_2) is independent. So, for simplicity of description, we ignore the part in the following security proof. That is, $z_i \leftarrow (z^*)^{u_1^{(i)}}(w^*)^{u_2^{(i)}}$ and $w_i \leftarrow (z^*)^{v_1^{(i)}}(w^*)^{v_2^{(i)}}$.

Then $(z^*, w^*, \sigma^*, \sigma_i)$ are expressed by the following equations:

$$\begin{aligned}
&\log_{g_1} z^* \equiv x_1^* + \eta x_2^* \pmod p \\
&\log_{g_1} w^* \equiv y_1^* + \eta y_2^* \pmod p \\
&\log_{g_1} \sigma^* \equiv r_1^*(x_1^* + d^* y_1^*) + \eta r_2^*(x_2^* + d^* y_2^*) \pmod p \\
&\log_{g_1} \sigma_i \equiv r_1^{(i)}((u_1^{(i)} + v_1^{(i)} d_i)x_1^* + (u_2^{(i)} + v_2^{(i)} d_i)y_1^*) + \\
&\qquad\qquad \eta r_2^{(i)}((u_1^{(i)} + v_1^{(i)} d_i)x_2^* + (u_2^{(i)} + v_2^{(i)} d_i)y_2^*) \pmod p.
\end{aligned}$$

where $g_2 = g_1^{\eta}$, $C_1^* = g_1^{r_1^*}$, $C_2^* = g_1^{r_2^*}$, $C_1^{(i)} = g_1^{r_1^{(i)}}$, $C_2^{(i)} = g_1^{r_2^{(i)}}$.

$$\begin{bmatrix}
1 & \eta & 0 & 0 \\
0 & 0 & 1 & \eta \\
r_1^* & \eta r_2^* & d^* r_1^* & \eta d^* r_2^* \\
r_1^{(i)}(u_1^{(i)} + v_1^{(i)} d_i) & \eta r_2^{(i)}(u_1^{(i)} + v_1^{(i)} d_i) & r_1^{(i)}(u_2^{(i)} + v_2^{(i)} d_i) & \eta r_2^{(i)}(u_2^{(i)} + v_2^{(i)} d_i)
\end{bmatrix}. \tag{4}$$

This matrix (4) is regular if and only if

$$\eta^2 (r_2^* - r_1^*)(r_2^{(i)} - r_1^{(i)})(d_i(d^* v_1^{(i)} - v_2^{(i)}) + (d^* u_1^{(i)} - u_2^{(i)})) \not\equiv 0 \pmod p. \tag{5}$$

$\eta \neq 0$ and $r_2^* - r_1^* \neq 0$, since we assume that $(\mathbb{G}, g_1, g_2, X_1^*, X_2^*) \notin \mathbb{D}(k)$ and $g_1 \neq 1, g_2 \neq 1, g_1 \neq g_2$. Since we are now considering Case (ii), $r_2^{(i)} - r_1^{(i)} \neq 0$, and $d_i(d^* v_1^{(i)} - v_2^{(i)}) + (d^* u_1^{(i)} - u_2^{(i)}) \not\equiv 0 \pmod p$ by the special rejection rule in game $\mathbf{G}_3'$.

Hence, this matrix (4) is regular. So, the remaining part of the proof is exactly the same as that of Claim 16 (Claim 11). □

Summing up Claims 13, 14, 18 and 19, we obtain the following claim,

Claim 20. *For any adversary $\mathcal{A}$ in the IND-CPCA game there exist probabilistic machines, $\mathcal{A}_1', \mathcal{A}_2'$ and $\mathcal{A}_3'$ whose running times are at most that of $\mathcal{A}$, such that*

$$\begin{aligned}
&\mathsf{Adv}\mathrm{KEM}_{\mathcal{A}}^{\textit{IND-CPCA}}(k) \leq \\
&\qquad \mathsf{Adv}\mathrm{DDH}_{\mathcal{A}_1'}(k) + t(n) \cdot \mathsf{Adv}\mathrm{GTCR}_{\mathsf{H},\mathcal{A}_2'}^{\mathsf{Conv},\mathsf{Rel}}(k) + \mathsf{Adv}\pi\mathrm{PRF}_{\mathsf{F},I_{\mathbb{G}},\mathcal{A}_3'}(k) + 4/p.
\end{aligned}$$

□

5 Conclusion and Open Problems

This paper presented a paradigm to design cryptographic primitives without random oracles under three assumptions: the decisional Diffie-Hellman (DDH) assumption, target collision resistant (TCR) hash function family (or GTCR hash function family) and a class of pseudo-random function (PRF) family, πPRF family.

The most important open problem in this paradigm is how to construct a πPRF family from a fundamental cryptographic primitive like a one-way function or (trapdoor) one-way permutation. Another important open problem is to clarify the relationship (or equivalence) between the CPCA-security and complete non-malleability [4].

Acknowledgments

The author would like to thank Dennis Hofheinz and Eike Kiltz for giving me insightful comments on PRFs and πPRFs, especially a result that proves the impossibility of our solution with using a plain PRF in a blackbox manner [7]. I would like to thank Masayuki Abe for his invaluable discussions and suggestions on the proposed paradigm, KEM and redundancy-free hybrid encryption. I also thank Alfred Menezes for his lectures on the AKE protocols and security models as well as insightful discussions and comments. I am also grateful to Michel Abdalla, Goichiro Hanaoka, Ran Canetti and David Pointcheval for their valuable comments and suggestions on a preliminary version of this manuscript. I would like to express my appreciation of Kaoru Kurosawa, the PC chair of Asiacrypt 2007, for giving me an opportunity to present this result at Asiacrypt 2007 as an invited talk.

References

1. Abe, M., Gennaro, R., Kurosawa, K. and Shoup, V., Tag-KEM/DEM: A New Framework for Hybrid Encryption and New Analysis of Kurosawa-Desmedt KEM, Adv. in Cryptology – Eurocrypt 2005, LNCS 3494, pp. 128-146 (2005).
2. Canetti, R. and Krawczyk, H., Analysis of key-exchange protocols and their use for building secure channels, Advances in Cryptology, EUROCRYPT 2001, LNCS 2045 (2001), http://eprint.iacr.org/2001/040.
3. Cramer, R. and Shoup, V., Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. SIAM Journal on Computing, 33(1), 167-226, (2003).
4. Fischlin, M., Completely Non-Malleable Schemes, Proceedings of ICALP 2005, LNCS 3580, Springer-Verlag, pp. 779-790 (2005).
5. Goldreich, O. , Goldwasser, S. and Micali, S.. How to Construct Random Functions. In Journal of the ACM, vol.33, no.4, pp.792-807 (1986).
6. Hastad, J. , Impagliazzo, R., Levin, L. and Luby, M., A Pseudorandom Generator from any One-way Function. SIAM Journal on Computing, v28 n4, pp.-1364-1396 (1999).
7. Hofheinz, D., private communication (2007).
8. Krawczyk, H., HMQV: A high-performance secure Diffie-Hellman protocol, Advances in Cryptology, CRYPTO 2005, LNCS 3621 (2005), http://eprint.iacr.org/2005/176.
9. Kurosawa, K. and Desmedt, Y., A New Paradigm of Hybrid Encryption Scheme, Advances in Cryptology- CRYPTO 2004, LNCS 3152, Springer-Verlag, pp. 426-442 (2004).
10. LaMacchia, B., Lauter, K. and Mityagin, A., Stronger security of authenticated key exchange, Cryptology ePrint Archive, Report 2006/073, 2006, http://eprint.iacr.org/2006/073.
11. Law, L., Menezes, A., Qu, M., Solinas, J. and Vanstone, S., An efficient protocol for authenticated key agreement, Designs, Codes and Cryptography 28, pp.119–134 (2003),
12. Menezes, A., Another look at HMQV, Journal of Mathematical Cryptology 1, pp.148–175 (2007),

13. Matsumoto, T., Takashima, Y. and Imai, H., On Seeking Smart Public-key Distribution Systems. Transactions of the IECE of Japan, E69:99-106 (1986).
14. Naor, M. and Yung, M., Universal one-way hash functions and their cryptographic applications. In Proceedings of the 21st Annual ACM Symposium on Theory of Computing, pp.33-43 (1989.)
15. Okamoto, T., Authenticated Key Exchange and Key Encapsulation in the Standard Model, Advances in Cryptology- Asiacrypt 2007, LNCS, Springer-Verlag (2007).
16. Rompel, J., One-way functions are necessary and sufficient for secure signatures. In Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, pp.387-394 (1990)
17. Ustaoglu, B., Obtaining a secure and efficient key agreement protocol from (H)MQV and NAXOS, Cryptology ePrint Archive, Report 2006/073, 2006, http://eprint.iacr.org/2007/123.