A New Paradigm of Hybrid Encryption Scheme

Kaoru Kurosawa[1] and Yvo Desmedt[2]

[1] Ibaraki University, Japan
kurosawa@cis.ibaraki.ac.jp
[2] Dept. of Computer Science, University College London, UK, and
Florida State University (USA)

Abstract. In this paper, we show that a key encapsulation mechanism (KEM) does not have to be IND-CCA secure in the construction of hybrid encryption schemes, as was previously believed. That is, we present a more efficient hybrid encryption scheme than Shoup [12] by using a KEM which is not necessarily IND-CCA secure. Nevertheless, our scheme is secure in the sense of IND-CCA under the DDH assumption in the standard model. This result is further generalized to universal$_2$ projective hash families.

Keywords: hybrid encryption, KEM, standard model

1 Introduction

1.1 Background

Cramer and Shoup showed the first provably secure practical public-key encryption scheme in the standard model [3, 6]. It is secure against adaptive chosen ciphertext attack (IND-CCA) under the Decisional Diffie-Hellman (DDH) assumption. They further generalized their scheme to projective hash families [4]. (In the random oracle model [1], many practical schemes have been proven to be IND-CCA, for example, OAEP+ [13], SAEP [2] , RSA-OAEP [8], etc. [7]. However, while the random oracle model is a useful tool, it does not rule out all possible attacks.)

On the other hand, a hybrid encryption scheme uses public-key encryption techniques to derive a shared key that is then used to encrypt the actual messages using symmetric-key techniques.

For hybrid encryption schemes, Shoup formalized the notion of a key encapsulation mechanism (KEM), and an appropriate notion of security against adaptive chosen ciphertext attack [12, 6]. A KEM works just like a public key encryption scheme, except that the encryption algorithm takes no input other than the recipient's public key. The encryption algorithm can only be used to generate and encrypt a key for a symmetric-key encryption scheme. (One can always use a public-key encryption scheme for this purpose. However, one can construct a KEM in other ways as well.) A secure KEM, combined with an appropriately secure symmetric-key encryption scheme, yields a hybrid encryption scheme which is secure in the sense of IND-CCA [12].

Shoup presented a secure KEM under the DDH assumption [12]. As a result, his hybrid encryption scheme is secure in the sense of IND-CCA under the DDH assumption in the standard model [12].

1.2 Our Contribution

In order to prove the security of hybrid encryption schemes, one has believed that it is essential for KEM to be secure in the sense of IND-CCA, as stated in [6, Remark 7.2, page 207].

In this paper, however, we disprove this belief. That is, it is shown that KEM does not have to be CCA secure, as was previously believed. On a more concrete level, we present a more efficient hybrid encryption scheme than Shoup [12] by using a KEM which is not necessarily secure in the sense of IND-CCA. Nevertheless, we prove that the proposed scheme is secure in the sense of IND-CCA under the DDH assumption in the standard model.

In a typical implementation, the underlying Abelian group may be a subgroup of Z_p^*, where p is a large prime. In this case, the size of our ciphertexts is $|p|$ bits shorter than that of Shoup [12]. The number of exponentiations per encryption and that of per decryption are also smaller. (Further, our scheme is more efficient than the basic Cramer-Shoup scheme [3, 6].)

This shows that one can start with a weak KEM and repair it with a hybrid construction. Eventually, more efficient hybrid encryption schemes could be obtained.

Our KEM is essentially a universal$_2$ projective hash family [4]. We present a generalization of our scheme to universal$_2$ projective hash families also.

The only (conceptual) cost one pays is that one needs to assume a simple condition on the symmetric encryption scheme. Namely, any fixed ciphertext is rejected with overwhelming probability, where the probability is taken over keys K. This property is already satisfied by the symmetric encryption scheme SKE which is used in the hybrid construction of Shoup [12]. Hence the SKE can be used in our hybrid construction too.

Our result gives new light to Cramer-Shoup encryption schemes [3, 4, 6] and opens a door to design more efficient hybrid encryption schemes.

2 Preliminaries

We denote by λ a security parameter. PPT denotes probabilistic polynomial time.

2.1 Notation and Definitions

$|S|$ denotes the cardinality of S if S is a set. $|m|$ denotes the bit length of m if m is a string or a number. If $A(\cdot, \cdot, \cdots)$ is a probabilistic algorithm, then $x \stackrel{R}{\leftarrow} A(x_1, x_2, \cdots)$ denotes the experiment of running A on input $x_1, x_2, \cdots$ and letting x be the outcome. If S is a set, $x \stackrel{R}{\leftarrow} S$ denotes the experiment of choosing $x \in S$ at random.

2.2 Public-Key Encryption Scheme (PKE)

A public-key encryption scheme is a three tuple of algorithms $\mathsf{PKE} = (\mathcal{K}_p, \mathcal{E}_p, \mathcal{D}_p)$. The key generation algorithm $\mathcal{K}_p$ generates a pair $(pk, sk) \stackrel{R}{\leftarrow} \mathcal{K}_p(1^\lambda)$, where pk is a public key and sk is a secret key. The encryption algorithm $\mathcal{E}_p$ takes a public key pk and a plaintext m, and returns a ciphertext $c \stackrel{R}{\leftarrow} \mathcal{E}_p(pk, m)$. The decryption algorithm $\mathcal{D}_p$ takes a secret key sk and a ciphertext c, and returns m or $reject$.

The chosen plaintext attack (IND-CPA) game is defined as follows. We imagine a PPT adversary A that runs in two stages. In the "find" stage, A takes a public key pk and queries a pair of equal length messages m_0 and m_1 to an encryption oracle. The encryption oracle chooses $b \stackrel{R}{\leftarrow} \{0, 1\}$ and computes a challenge ciphertext c^* of m_b randomly. In the "guess" stage, given c^*, A outputs a bit $\tilde{b}$ and halts.

The adaptive chosen ciphertext attack (IND-CCA) game is defined similarly. The difference is that the adversary A is given access to a decryption oracle, where A cannot query the challenge ciphertext c^* itself in the guess stage.

Definition 1. *We say that* PKE *is secure in the sense of IND-CCA if* $|\Pr(\tilde{b} = b) - 1/2|$ *is negligible in the IND-CCA game for any PPT adversary* A*.*

In particular, we define the IND-CCA advantage of A as follows:

$$\mathtt{Adv}^{cca}_{\mathsf{PKE}}(A) = |\Pr(\tilde{b} = b) - 1/2|. \tag{1}$$

For any t and q_d, define $\mathtt{Adv}^{cca}_{\mathsf{PKE}}(t, q_d) = \max_A \mathtt{Adv}^{cca}_{\mathsf{PKE}}(A)$, where the maximum is taken over all A which runs in time t and makes at most q_d queries to the decryption oracle.

2.3 Diffie-Hellman Assumptions

Let G be an Abelian group of order Q, where Q is a large prime. Let g_1 be a generator of G. Let

$$\begin{aligned} DH &= \{(g_1, g_2, g_1^r, g_2^r) \mid r \in Z_Q, g_2 = g_1^w, w \in Z_Q\} \\ Random &= \{(g_1, g_2, g_1^{r_1}, g_2^{r_2}) \mid r_1 \in Z_Q, r_2 \in Z_Q, g_2 = g_1^w, w \in Z_Q\} \end{aligned}$$

The decisional Diffie-Hellman (DDH) assumption claims that DH and $Random$ are indistinguishable.

For a distinguisher D, consider the following two experiments. In experiment 0, let $(g_1, g_2, u_1, u_2) \stackrel{R}{\leftarrow} DH$. In experiment 1, let $(g_1, g_2, u_1, u_2) \stackrel{R}{\leftarrow} Random$. Define

$$\mathtt{Adv}^{ddh}_{\mathsf{G}}(D) \stackrel{\triangle}{=} |p_0 - p_1|,$$

where

$$p_0 \stackrel{\triangle}{=} \Pr(D = 1 \text{ in experiment } 0),\ p_1 \stackrel{\triangle}{=} \Pr(D = 1 \text{ in experiment } 1)$$

For any t, define $\mathtt{Adv}^{ddh}_{\mathsf{G}}(t) \stackrel{\triangle}{=} \max_A \mathtt{Adv}^{ddh}_{\mathsf{G}}(D)$, where the maximum is taken over all D which runs in time t.

2.4 Target Collision Resistant Hash Function

The notion of target collision resistant TCR family of hash functions was shown by Cramer and Shoup [6]. It is a special case of universal one-way hash function UOWH family introduced by Naor and Yung [10], where a UOWH family can be built from arbitrary one-way functions [10, 11].

In a TCR family, given a randomly chosen tuple of group elements x ($\in G^n$ for some n) and a randomly chosen hash function H, it is infeasible for an adversary A to find $y \neq x$ such that $H(x) = H(y)$. (In a UOWH family, x is chosen by the adversary.) In practice, one can use a dedicated cryptographic hash function, like SHA-1. Define

$$\mathtt{Adv}^{hash}_{\mathsf{TCR}}(A) \stackrel{\triangle}{=} \Pr(A \text{ succeeds}).$$

For any t, define $\mathtt{Adv}^{hash}_{\mathsf{TCR}}(t) \stackrel{\triangle}{=} \max_A \mathtt{Adv}^{hash}_{\mathsf{TCR}}(A)$, where the maximum is taken over all A which runs in time t.

3 Previous Results on KEM

It is known that by combining a KEM and a one-time symmetric encryption scheme which are both secure in the sense of IND-CCA, we can obtain a hybrid encryption scheme which is secure in the sense of IND-CCA.

3.1 KEM [12][6, Sec.7.1]

A key encapsulation mechanism KEM consists of the following algorithms.

- A key generation algorithm $\mathsf{KEM.Gen}$ that on input 1^λ outputs a public/secret key pair (pk, sk).
- An encryption algorithm $\mathsf{KEM.Enc}$ that on input 1^λ and a public key pk, outputs a pair (K, ψ), where K is a key and ψ is a ciphertext.
  A key K is a bit string of length $\mathsf{KEM.Len}(\lambda)$, where $\mathsf{KEM.Len}(\lambda)$ is another parameter of KEM.
- A decryption algorithm $\mathsf{KEM.Dec}$ that on input 1^λ, a secret key sk, a string (in particular a ciphertext) ψ, outputs either a key K or the special symbol reject.

$\mathsf{KEM.Gen}$ and $\mathsf{KEM.Enc}$ are PPT algorithms and $\mathsf{KEM.Dec}$ is a deterministic polynomial time algorithm.

In the chosen ciphertext attack (IND-CCA) game, we imagine a PPT adversary A that runs in two stages. In the find stage, A takes a public key pk and queries an encryption oracle. The encryption oracle computes:

$$(K^*, \psi^*) \stackrel{R}{\leftarrow} \mathsf{KEM.Enc}(1^\lambda); K^+ \stackrel{R}{\leftarrow} \{0,1\}^k; \tau \stackrel{R}{\leftarrow} \{0,1\};$$

$$\text{if } \tau = 0 \text{ then } K^\dagger \leftarrow K^* \text{ else } K^\dagger \leftarrow K^+$$

where $k = \mathsf{KEM.Len}(\lambda)$, and responds with the pair $(K^\dagger, \psi^*)$. In the guess stage, given $(K^\dagger, \psi^*)$, the adversary A outputs a bit $\tilde{\tau}$ and halts.

The adversary A is also given access to a decryption oracle. For each decryption oracle query, the adversary A submits a ciphertext ψ, and the decryption oracle responds with $\mathsf{KEM.Dec}(1^\lambda, sk, \psi)$, where A cannot query the challenge ciphertext ψ^* itself in the guess stage.

Definition 2. *We say that* KEM *is secure in the sense of IND-CCA if* $|\Pr(\tilde{\tau} = \tau) - 1/2|$ *is negligible in the above game for any PPT adversary A.*

3.2 One-Time Symmetric-Key Encryption [6, Sec.7.2]

A one-time symmetric-key encryption scheme SKE consists of two algorithms:

- A deterministic polynomial time encryption algorithm SKE.Enc that takes as input 1^λ, a key K and a message m, and outputs a ciphertext χ.
- A deterministic polynomial time decryption algorithm SKE.Dec that takes as input 1^λ, a key K and a ciphertext χ, and outputs a message m or the special symbol reject.

The key K is a bit string of length $\mathsf{SKE.Len}(\lambda)$, where $\mathsf{SKE.Len}(\lambda)$ is a parameter of the encryption scheme.

In the passive attack game, we imagine a PPT adversary A that runs in two stages. In the "find" stage, A takes 1^λ, and queries a pair of equal length messages m_0 and m_1 to an encryption oracle. The encryption oracle generates a random key K of length $\mathsf{SKE.Len}(\lambda)$, along with random $\sigma \stackrel{R}{\leftarrow} \{0,1\}$, and encrypts m_σ using the key K. In the "guess" stage, given the resulting ciphertext χ^*, A outputs a bit $\tilde{\sigma}$ and halts.

In the chosen ciphertext attack (IND-CCA) model, the adversary A is also given access to a decryption oracle in the guess stage. In each decryption oracle query, A submits a ciphertext $\chi \neq \chi^*$, and obtains the decryption of χ under the key K.

Definition 3. *We say that* SKE *is secure in the sense of IND-CCA if* $|\Pr(\tilde{\sigma} = \sigma) - 1/2|$ *is negligible in the IND-CCA game for any PPT adversary A.*

In particular, we define the IND-CCA advantage of A as follows.

$$\mathtt{Adv}^{cca}_{\mathsf{SKE}}(A) = |\Pr(\tilde{\sigma} = \sigma) - 1/2|. \tag{2}$$

For any t and q_d, define $\mathtt{Adv}^{cca}_{\mathsf{SKE}}(t, q_d) = \max_A \mathtt{Adv}^{cca}_{\mathsf{SKE}}(A)$, where the maximum is taken over all A which runs in time t and makes at most q_d queries to the decryption oracle.

3.3 Construction of SKE

Shoup showed a construction of a one-time symmetric-key encryption scheme as follows [12, page 281]. Let PRBG be a pseudo-random bit generator which stretches l-bit strings to strings of arbitrary (polynomial) length. We assume that $1/2^l$ is a negligible quantity. In a practical implementation, it is perfectly reasonable to stretch the key K_0 by using it as the key to a dedicated block cipher, and then evaluate the block cipher at successive points (so called "counter mode") to obtain a sequence of pseudo-random bits [6, Sec.7.2.2].

Let AXUH be a hash function which is suitable for message authentication, i.e., an almost XOR-universal hash function [9]. We assume that AXUH is keyed by an l'-bit string and hashes arbitrary bit string to l-bit strings. Many efficient constructions for AXUH exist that do not require any intractability assumptions.

To encrypt a message m by using a key $K = (K_0, K_1, K_2)$, we apply PRBG to K_0 to obtain an $|m|$-bit string f. Then we compute

$$e = f \oplus m, \tag{3}$$

$$a = \mathsf{AXUH}(K_1, e) \oplus K_2. \tag{4}$$

The ciphertext is $\chi = (e, a)$, where a is called a tag. (We can generate K by applying PRBG to a shorter key.)

To decrypt $\chi = (e, a)$ using a key $K = (K_0, K_1, K_2)$, we first test if eq.(4) holds. If it does not hold, then we *reject*. Otherwise, we output $m = e \oplus f$.

3.4 A Hybrid Construction

Let KEM be a key encapsulation mechanism and let SKE be a one-time symmetric key encryption scheme such that $\mathsf{KEM.Len}(\lambda) = \mathsf{SKE.Len}(\lambda)$ for all λ. Let HPKE be the hybrid public-key encryption scheme obtained from KEM and SKE.

Proposition 1. *[6, Theorem 7.2] If* KEM *and* SKE *are secure in the sense of IND-CCA, then so is* HPKE.

4 Proposed Hybrid Encryption Scheme

In this section, we show a more efficient hybrid encryption scheme than before [12, 6] by using a KEM which is not necessarily secure in the sense of IND-CCA. Nevertheless, we prove that the proposed scheme is secure in the sense of IND-CCA under the DDH assumption in the standard model.

4.1 Overview

A KEM works just like a public key encryption scheme, except that the encryption algorithm takes no input other than the recipient's public key. Instead, the encryption algorithm generates a pair (K, ψ), where K is a key of SKE and ψ is an encryption of K. The decryption algorithm applied to ψ yields K. In our hybrid encryption scheme, $\psi = (u_1, u_2) = (g_1^r, g_2^r)$.

The notion of IND-CCA is adapted to KEM as follows. The adversary does not give two messages to the encryption oracle. Rather, the encryption oracle runs the KEM encryption algorithm to obtain a pair (K, ψ). The encryption oracle then gives the adversary either (K, ψ) or (K^+, ψ), where K^+ is an independent random bit string; the choice of K versus K^+ depends on the value of the random bit b chosen by the encryption oracle.

Up to now, in order to prove the security of the hybrid encryption scheme, it has been believed to be essential for KEM to be secure in the sense of IND-CCA, as stated in [6, Remark 7.2, page 207].

However, we know of no way to prove that our KEM is secure in the sense of IND-CCA. Nevertheless, we prove that the proposed hybrid encryption scheme is secure in the sense of IND-CCA. This shows that one can start with a weak KEM and repair it with a hybrid construction. Eventually, more efficient hybrid encryption schemes could be obtained.

A generalization of our scheme to universal$_2$ projective hash families [4] will be given in Sec.8.

4.2 ϵ-Rejection Secure

We require that a one-time symmetric-key encryption scheme SKE satisfies the following property: any bit string χ is rejected by the decryption algorithm with overwhelming probability. Formally, we say that SKE is ϵ-rejection secure if for any bit string χ,

$$\Pr(\mathsf{SKE.Dec}(1^\lambda, K, \chi) = reject) \geq 1 - \epsilon,$$

where the probability is taken over K.

This property is already satisfied by the one-time symmetric-key encryption scheme shown in Sec.3.3. Indeed, for any fixed $\chi = (e, a)$, eq.(4) holds with probability $1/2^l$ because K_2 is random. Therefore, this encryption scheme is ϵ-rejection secure for $\epsilon = 1/2^l$.

4.3 Proposed Scheme

The proposed hybrid encryption scheme is based on the basic Cramer-Shoup scheme [3, 6]. However, it does not use v as the validity check as in [3, 6], but rather it is used to derive the encapsulated key K. This saves the value h which was previously used to encapsulate the key, and one exponentiation encryption/decryption. It also makes the public key and the secret key one element shorter.

Let G be an Abelian group of order Q, where Q is a large prime. Let SKE be a one-time symmetric-key encryption scheme.

Let $H : G \to \{0, 1\}^k$ be a hash function, where $k = \mathsf{SKE.Len}(\lambda)$. We assume that $H(v)$ is uniformly distributed over $\{0, 1\}^k$ if v is uniformly distributed over G. This is a very weak requirement on H, and we can use SHA-1, for example.

Key Generation. Generate two distinct generators g_1, g_2 of G at random. Choose $(x_1, x_2, y_1, y_2) \in Z_Q^4$ at random. Compute

$$c = g_1^{x_1} g_2^{x_2}, \ d = g_1^{y_1} g_2^{y_2}.$$

Finally, a random κ indexing a target collision resistant hash function TCR (see Sec.2.4) is chosen. The public-key is $pk = (g_1, g_2, c, d, \kappa)$ and the secret key is $sk = (x_1, x_2, y_1, y_2)$.

Encryption. To encrypt a message m, choose $r \in Z_Q$ at random and compute

$$u_1 = g_1^r,\ u_2 = g_2^r, \alpha = \mathsf{TCR}(\kappa; u_1, u_2),$$

$$v = c^r d^{r\alpha},\ K = H(v),\ \chi = \mathsf{SKE.Enc}(1^\lambda, K, m).$$

The ciphertext is (u_1, u_2, χ). (In the ciphertext, the KEM part is $\psi = (u_1, u_2)$.)

Decryption. For a ciphertext $C = (u_1, u_2, \chi)$, compute

$$\alpha = \mathsf{TCR}(\kappa; u_1, u_2),\ v = u_1^{x_1 + y_1 \alpha} u_2^{x_2 + y_2 \alpha},\ K = H(v).$$

Then decrypt χ under K using $\mathsf{SKE.Dec}$, and output the resulting decryption z. (z may be reject.)

4.4 Security

Theorem 1. The proposed hybrid encryption scheme Hybrid is secure in the sense of IND-CCA under the DDH assumption if SKE is secure in the sense of IND-CCA and it is ϵ-rejection secure for negligible ϵ. In particular,

$$\mathtt{Adv}^{cca}_{\mathsf{Hybrid}}(t, q_d) \le \mathtt{Adv}^{ddh}_{\mathsf{G}}(t_1) + \mathtt{Adv}^{hash}_{\mathsf{TCR}}(t_2) + \mathtt{Adv}^{cca}_{\mathsf{SKE}}(t_3, q_d) + q_d(\epsilon + \frac{1}{Q}) + \frac{4}{Q}.$$

where t_1, t_2, t_3 are essentially the same as t.

A proof will be given in the next section.

4.5 Efficiency Comparison

In the hybrid encryption scheme of Shoup [12] and in the Cramer-Shoup scheme [3],

- $v \in G$ is included in the ciphertext C to check the validity of C.
- $h \in G$ is included in a public-key to generate a key K of SKE.

In our scheme, on the other hand,

- v is not included in the ciphertext, but it is used to derive a key K of SKE.
- h is not necessary at all.

In a typical implementation, the underlying Abelian group G may be a subgroup of Z_p^*, where p is a large prime. Table 1 shows an efficiency comparison among the proposed hybrid encryption scheme, the hybrid encryption scheme of Shoup [12] and the basic Cramer-Shoup scheme [3]. (In the table, a denotes the tag of SKE as shown in Sec.3.3.)

We can see that

– the size of our ciphertext is $|p|$ bits shorter than that of Shoup [12].
– the size of our public-key is $|p|$ bits shorter than that of Shoup [12].
– The number of exponentiations per encryption and that of per decryption of our scheme are also smaller.

Further, our scheme is more efficient than the Cramer-Shoup scheme [3] for $|m| < 2|p| - |a|$. Moreover, in Cramer-Shoup [3] m must belong to G (so $|m| \leq |q|$), while in ours and Shoup's [12] $m \in \{0,1\}^*$ (polynomial length).

Table 1. Efficiency Comparison

	ciphertext	public-key	exp/enc	exp/dec
Cramer-Shoup [3]	$4 \cdot \|p\|$	$5\|p\| + \|\kappa\|$	5	3
Shoup [12]	$3 \cdot \|p\| + \|m\| + \|a\|$	$5\|p\| + \|\kappa\|$	5	3
Proposed	$2 \cdot \|p\| + \|m\| + \|a\|$	$4\|p\| + \|\kappa\|$	4	2

5 Proof of Theorem 1

5.1 Outline

The following lemma is simple but useful.

Lemma 1. *[6, Lemma 6.2] Let S_1, S_2 and F be events defined on some probability space. Suppose that the event $S_1 \vee \neg F$ occurs if and only if $S_2 \vee \neg F$ occurs. Then*

$$|\Pr(S_1) - \Pr(S_2)| \leq \Pr(F).$$

Let A be an adversary who breaks the proposed scheme in the sense of IND-CCA. The attack game is as described in Sec.2.2. Suppose that the public key is (g_1, g_2, c, d, κ) and the secret key is (x_1, x_2, y_1, y_2). The target ciphertext is denoted by $C^* = (u_1^*, u_2^*, \chi^*)$. We also denote by r^*, α^*, v^*, K^* the values corresponding with r, α, v, K related to C^*.

Suppose that A queries at most q_1 times to the decryption oracle in the find stage, and at most q_2 times to the decryption oracle in the guess stage, where $q_d = q_1 + q_2$. We say that a ciphertext $C = (u_1, u_2, \chi)$ is valid if $u_1 = g_1^r$ and $u_2 = g_2^r$ for some r. Otherwise, we say that C is invalid.

Let $\log(\cdot)$ denote $\log_{g_1}(\cdot)$ and let $w = \log g_2$. Then

$$\log c = x_1 + w x_2 \tag{5}$$

$$\log d = y_1 + w y_2 \tag{6}$$

Let $\mathbf{G_0}$ be the original attack game, let $\tilde{b} \in \{0,1\}$ denote the output of A, and let T_0 be the event that $b = \tilde{b}$ in $\mathbf{G_0}$. Therefore,

$$\mathtt{Adv}^{cca}_{\mathsf{Hybrid}}(A) = |\Pr[T_0] - 1/2|.$$

We shall define a sequence $\mathbf{G_1}, \cdots, \mathbf{G}_\ell$ of modified attack games. For any $1 \leq i \leq \ell$, we let T_i be the event that $b = \tilde{b}$ in game $\mathbf{G_i}$.

In game $\mathbf{G_1}$, we modify the encryption oracle as follows: $v^* = c^{r^*} d^{r^* \alpha^*}$ is replaced by

$$v^* = (u_1^*)^{x_1 + y_1 \alpha^*} (u_2^*)^{x_2 + y_2 \alpha^*}.$$

This change is purely conceptual, and $\Pr[T_1] = \Pr[T_0]$.

In game $\mathbf{G_2}$, we modify the encryption oracle again, so that (u_1^*, u_2^*) is replaced by a random pair $(g_1^{r_1^*}, g_2^{r_2^*})$ with $r_1^* \neq r_2^*$. Under the DDH assumption, A will hardly notice, and $|\Pr[T_2] - \Pr[T_1]|$ is negligible. More precisely, we have

Lemma 2. *There exists a PPT algorithm A_1, whose running time is essentially the same as that of A, such that*

$$|\Pr[T_2] - \Pr[T_1]| \leq \mathtt{Adv}_{\mathsf{G}}^{ddh}(A_1) + 3/Q.$$

The proof is the same as that of [6, Lemma 6.3].

In game $\mathbf{G_3}$, we modify the decryption oracle, so that it applies the following special rejection rule: In the guess stage, if the adversary submits a ciphertext $(u_1, u_2) \neq (u_1^*, u_2^*)$ but $\alpha = \alpha^*$, then the decryption oracle immediately outputs reject and halts. Let R_3 be the event that the decryption oracle in game $\mathbf{G_3}$ rejects a ciphertext using the special rejection rule. It is clear that games $\mathbf{G_2}$ and $\mathbf{G_3}$ proceed identically until the event R_3 occurs. In particular, the event $T_2 \wedge \neg R_3$ and $T_3 \wedge \neg R_3$ are identical. So by Lemma 1, we have

$$|\Pr[T_3] - \Pr[T_2]| \leq \Pr[R_3].$$

Lemma 3. *There exists a PPT algorithm A_2, whose running time is essentially the same as that of A, such that*

$$\Pr[R_3] \leq \mathtt{Adv}_{\mathsf{TCR}}^{hash}(A_2) + 1/Q.$$

The proof is the same as that of [6, Lemma 6.5].

In game $\mathbf{G_4}$, we modify the decryption oracle, so that it rejects all invalid ciphertexts C in the find stage. Let R_4 be the event that a ciphertext is rejected in $\mathbf{G_4}$ that would not have been rejected under the rules of game $\mathbf{G_3}$. It is clear that games $\mathbf{G_3}$ and $\mathbf{G_4}$ proceed identically until the event R_4 occurs. In particular, the event $T_3 \wedge \neg R_4$ and $T_4 \wedge \neg R_4$ are identical. So by Lemma 1, we have

$$|\Pr[T_4] - \Pr[T_3]| \leq \Pr[R_4].$$

Lemma 4. $\Pr[R_4] \leq q_1 \cdot \epsilon$. *(For the proof, see Section 5.2.)*

In game $\mathbf{G_5}$, we modify the encryption oracle as follows. $(u_1^*, u_2^*) = (g_1^{r_1^*}, g_2^{r_2^*})$ is randomly chosen in such a way that an event R_5 does not occur, where R_5 is the event that $(u_1^*, u_2^*) = (u_1, u_2)$ for some invalid ciphertext (u_1, u_2, χ) which A queries in the find stage. It is clear that the event $T_4 \wedge \neg R_5$ and $T_5 \wedge \neg R_5$ are identical. So by Lemma 1, we have

$$|\Pr[T_5] - \Pr[T_4]| \leq \Pr[R_5].$$

Lemma 5. $\Pr[R_5] \leq q_1/Q$. *(For the proof, see Section 5.3.)*

In game $\mathbf{G_6}$, we modify the decryption oracle, so that it rejects all invalid ciphertexts C in the guess stage. Let R_6 be the event that a ciphertext is rejected in $\mathbf{G_6}$ that would not have been rejected under the rules of game $\mathbf{G_5}$. It is clear that games $\mathbf{G_5}$ and $\mathbf{G_6}$ proceed identically until the event R_6 occurs. In particular, the event $T_5 \wedge \neg R_6$ and $T_6 \wedge \neg R_6$ are identical. So by Lemma 1, we have

$$|\Pr[T_6] - \Pr[T_5]| \leq \Pr[R_6].$$

Lemma 6. $\Pr[R_6] \leq q_2 \cdot \epsilon$. *(For the proof, see Section 5.4.)*

In game $\mathbf{G_7}$, we modify the encryption oracle and the decryption oracle, so that K^* is replaced by a random key K^+.

Lemma 7. $\Pr[T_6] = \Pr[T_7]$. *(For the proof, see Section 5.5.)*

Lemma 8. *There exists a PPT algorithm A_3, whose running time is essentially the same as that of A, such that*

$$\mathtt{Adv}^{cca}_{\mathsf{SKE}}(A_3) = |\Pr[T_7] - 1/2|.$$

For the proof, see Section 5.6.

From the above results, we immediately obtain that

$$\mathtt{Adv}^{cca}_{\mathsf{Hybrid}}(A_3) \leq \mathtt{Adv}^{ddh}_{\mathsf{G}}(A_1) + \mathtt{Adv}^{hash}_{\mathsf{TCR}}(A_2) + \mathtt{Adv}^{cca}_{\mathsf{SKE}}(A_3) + q_d(\epsilon + \frac{1}{Q}) + \frac{4}{Q}.$$

5.2 Proof of Lemma 4

From the A's view, (x_1, x_2, y_1, y_2) is a random point satisfying eq.(5) and eq.(6). Suppose that A queries an invalid ciphertext (u_1, u_2, χ) to the decryption oracle, where $\log_{g_1}(u_1) = r_1$ and $\log_{g_2}(u_2) = r_2$ with $r_1 \neq r_2$. Let $v = u_1^{x_1+y_1\alpha} u_2^{x_2+y_2\alpha}$, where $\alpha = \mathsf{TCR}(\kappa; u_1, u_2)$. Then

$$\log v = r_1(x_1 + \alpha y_1) + r_2 w(x_2 + \alpha y_2). \tag{7}$$

It is clear that eq.(5),(6) and (7) are linearly independent. This means that v can take any value. In other words, v is uniformly distributed over G. Hence $K = H(v)$ is uniformly distributed over $\{0,1\}^k$. Now since SKE is ϵ-rejection secure, the decryption oracle accepts (u_1, u_2, χ) with probability at most ϵ. Consequently, we obtain this lemma.

5.3 Proof of Lemma 5

For any fixed (u_1, u_2),

$$\Pr[(u_1^*, u_2^*) = (u_1, u_2)] = \frac{1}{Q(Q-1)} \leq 1/Q$$

because $(r_1^*, r_2^*) \in Z_Q^2$ is randomly chosen in such a way that $r_1^* \neq r_2^*$.

5.4 Proof of Lemma 6

As the worst case, we assume that A knows v^*. Then from the A's view, (x_1, x_2, y_1, y_2) is a random point satisfying eq.(5), (6) and

$$\log v^* = r_1^*(x_1 + \alpha^* y_1) + r_2^* w(x_2 + \alpha^* y_2). \tag{8}$$

In the guess stage, suppose that A queries an invalid ciphertext (u_1, u_2, χ) to the decryption oracle, where $\log u_1 = r_1$ and $\log u_2 = r_2$ with $r_1 \neq r_2$. Let $v = u_1^{x_1 + y_1 \alpha} u_2^{x_2 + y_2 \alpha}$, where $\alpha = \mathsf{TCR}(\kappa; u_1, u_2)$. Then

$$\log v = r_1(x_1 + \alpha y_1) + r_2 w(x_2 + \alpha y_2). \tag{9}$$

Now

$$\begin{vmatrix} 1 & 0 & w & 0 \\ 0 & 1 & 0 & w \\ r_1^* & \alpha^* r_1^* & w r_2^* & \alpha^* w r_2^* \\ r_1 & \alpha r_1 & w r_2 & \alpha w r_2 \end{vmatrix} = w^2(r_2^* - r_1^*)(r_2 - r_1)(\alpha - \alpha^*) \neq 0$$

Therefore, eq.(5), (6), (8) and (9) are linearly independent. This means that v is uniformly distributed over G. Hence $K = H(v)$ is uniformly distributed over $\{0,1\}^k$. Now since SKE is ϵ-rejection secure, the decryption oracle accepts (u_1, u_2, χ) with probability at most ϵ.

Consequently, we obtain this lemma.

5.5 Proof of Lemma 7

In game $\mathbf{G_6}$, from the A's view, (x_1, x_2, y_1, y_2) is a random point satisfying eq.(5) and eq.(6). Further, it is clear that eq.(5),(6) and (8) are linearly independent. This means that v^* can take any value. In other words, v^* is uniformly distributed over G. Hence $K^* = H(v^*)$ is uniformly distributed over $\{0,1\}^k$. Consequently, we obtain this lemma.

5.6 Proof of Lemma 8

We describe Algorithm A_3. Algorithm A_3 provides an environment for A as follows. First, A_3 runs the key generation algorithm of Hybrid to generate a public-key $pk = (g_1, g_2, c, d, \kappa)$ and the secret-key $sk = (x_1, x_2, y_1, y_2)$. In particular, A_3 chooses $w \in Z_Q$ randomly and computes $g_2 = g_1^w$. It then gives pk to A.

In the find stage, whenever A submits a ciphertext C to the decryption oracle, A_3 applies the decryption rule of game $\mathbf{G_7}$, using the secret-key sk and w.

When A submits (m_0, m_1) to the encryption oracle, A_3 submits (m_0, m_1) to her encryption oracle.

The encryption oracle of A_3 chooses a random key $K^+ \in \{0,1\}^k$ along with a random bit σ, and encrypts m_σ using the key K^+. It then returns the resulting ciphertext χ^* to A_3.

A_3 generates (u_1^*, u_2^*) according to the encryption rule of game $\mathbf{G_7}$. It then returns the target ciphertext $C^* = (u_1^*, u_2^*, \chi^*)$ to A.

In the guess stage, suppose that A submits a ciphertext $C = (u_1, u_2, \chi)$ to the decryption oracle. If $(u_1, u_2) \neq (u_1^*, u_2^*)$, then A_3 applies the decryption rule of game $\mathbf{G_7}$, using the secret-key sk and w. Otherwise, A_3 queries χ to her decryption oracle, where the decryption oracle decrypts χ by using K^+. A_3 then returns the answer to A.

When A outputs $\tilde{\sigma}$, A_3 outputs $\tilde{\sigma}$ and halts. That completes the description of A_3.

It is clear that A_3 perfectly simulates the environment of A. Therefore,

$$\Pr[T_7] = \Pr(\sigma = \tilde{\sigma}).$$

On the other hand,

$$\mathtt{Adv}_{\mathsf{SKE}}^{cca}(A_3) = |\Pr(\sigma = \tilde{\sigma}) - 1/2|.$$

Consequently, we obtain this lemma.

6 Discussion

We have argued that a KEM does not have to be CCA-secure in the construction of hybrid encryption schemes, as was previously believed.

In the IND-CCA definition of hybrid encryption schemes, the decryption oracle returns the message m for a queried ciphertext $C = (\psi, \chi)$, where ψ is the KEM part and χ is the symmetric encryption ciphertext. On the other hand, in the IND-CCA definition of KEM, the decryption oracle returns the symmetric key K for a queried ψ. Hence, the IND-CCA definition of KEM is too demanding because the decryption oracle reveals much more information than the decryption oracle of the hybrid encryption scheme does.

Then one may consider to define a weaker condition on KEM such that when coupled with CCA-secure symmetric encryption (with the extra condition of Section 3.4), it would yield a CCA-secure hybrid encryption scheme. However, it seems to be impossible because the security of KEM and that of the symmetric encryption scheme are intertwined (as in our scheme).

7 Hash Proof System

Cramer and Shoup introduced a notion of Hash Proof System (HPS) [4, 5] in order to generalize their encryption scheme based on the DDH assumption [3]. By using HPS, they showed new CCA-secure encryption schemes under Quadratic Residuosity assumption and Paillier's Decision Composite Residuosity assumption, respectively.

In this section, we give the definition of a slight variant of HPS, where ϵ-*universal*$_2$ is replaced by *strongly universal*$_2$.

7.1 Subset Membership Problem [4, 5]

A subset membership problem Mem specifies a collection $\{\mathsf{Instance}_n\}_{n\in\mathsf{N}}$ such that for every n, $\mathsf{Instance}_n$ is a probability distribution over problem instances Λ. Each Λ specifies the following:

- Define, non-empty sets, X, L and W such that $L \subset X$.
- A binary relation $R \subset X \times W$ such that $x \in L$ iff $(x, w) \in R$ for some witness $w \in W$.

We require that the following PPT algorithms exist.

1. *Instance sampling*: samples an instance Λ according to $\mathsf{Instance}_n$ on 1^n.
2. *Subset sampling*: outputs a random $x \in L$ together with a witness $w \in W$ for x on input 1^n and $\Lambda[X, L, W, R]$.
3. *Element sampling*: outputs a random $x \in X$.

We say that Mem is hard if (Λ, x_0) and (Λ, x_1) are indistinguishable for a random $x_0 \in L$ and a random $x_1 \in X \setminus L$.

7.2 Projective Hash Family

Let X and Π be finite, non-empty sets. Let $F = \{f_i : X \to \Pi\}_{i\in I}$ be a set of functions indexed by I. We call (F, I, X, Π) a universal hash family [4, 5].

Let $L \subset X$. Let S be a finite, non-empty set, and let $\alpha : I \to S$ be a function. Set $\mathsf{Project} = (F, I, X, L, \Pi, S, \alpha)$.

Definition 4. *[4, 5]* $\mathsf{Project} = (F, I, X, L, \Pi, S, \alpha)$ *is called a projective hash family if for all $i \in I$, the action of f_i on L is determined*[3] *by $\alpha(i)$.*

In other words, the value $f_i(x)$ is determined by $\alpha(i)$ if $x \in L$. We next define the notion of strongly universal$_2$ projective hash, a variant of Cramer-Shoup's ϵ-universal$_2$ projective hash.

Definition 5. Let $\mathsf{Project} = (F, I, X, L, \Pi, S, \alpha)$ be a projective hash family. Consider the probability space defined by choosing $i \in I$ at random. We say that $\mathsf{Project}$ is *strongly* universal$_2$ if

- for all $s \in S, x \in X \setminus L$, and $\pi \in \Pi$,
$$\Pr[f_i(x) = \pi \mid \alpha(i) = s] = 1/|\Pi|,$$
- and for all $s \in S, x, x^* \in X \setminus L$ with $x \neq x^*$, and $\pi, \pi^* \in \Pi$,
$$\Pr[f_i(x) = \pi \mid f_i(x^*) = \pi^* \wedge \alpha(i) = s] = 1/|\Pi|.$$

$\mathsf{Project}$ is strongly universal$_2$ means that for any $x \notin L$, the value of $f_i(x)$ is uniformly distributed over Π conditioned on a fixed value of $\alpha(i)$, and it is also uniformly distributed over Π conditioned on fixed values of $\alpha(i)$ and $f_i(x^*)$ for $x^* \notin L$ with $x^* \neq x$.

[3] For a further clarification, see Section 7.3.

7.3 Hash Proof System [4, 5]

Let Mem be a subset membership problem. A hash proof system (HPS) P for Mem associates with each instance $\Lambda[X, L, W, R]$ of Mem a projective hash family $\mathsf{Project} = (F, I, X, L, \Pi, S, \alpha)$.

P provides several algorithms to carry out basic operations: $i \stackrel{R}{\leftarrow} I$ and computing $\alpha(i) \in S$ given $i \in I$. The private evaluation algorithm for P computes $f_i(x) \in \Pi$ given $i \in I$ and $x \in X$. The public evaluation algorithm for P computes $f_i(x) \in \Pi$ given $\alpha(i) \in S, x \in X$ and $w \in W$, where w is a witness for x.

8 Proposed Hybrid Construction Based on HPS

In this section, we generalize our hybrid encryption scheme of Sec.4.3 by using the variant of HPS shown above. Then efficient hybrid encryption schemes are obtained which are secure in the sense of IND-CCA under Quadratic Residuosity assumption and Paillier's Decision Composite Residuosity assumption, respectively, in the standard model.

8.1 Hybrid Construction

Let Mem be a subset membership problem and P be a hash proof system for Mem. Let SKE be a one-time symmetric-key encryption scheme.

Key Generation. Generate an instance $\Lambda[X, L, W, R]$ using the instance sampling algorithm of Mem. Suppose that P associates with $\Lambda[X, L, W, R]$ a projective hash family $\mathsf{Project} = (F, I, X, L, \Pi, S, \alpha)$. Choose $i \in I$ at random and compute $s = \alpha(i)$.

The public key is s and the secret key is i. Let $H : \Pi \to \{0, 1\}^k$ be a hash function, where $k = \mathsf{SKE.Len}(\lambda)$. We assume that $H(v)$ is uniformly distributed over $\{0, 1\}^k$ if v is uniformly distributed over Π. This is a very weak requirement on H, and we can use SHA-1, for example.

Encryption. To encrypt a message m, generate $x \in L$ at random together with a witness $w \in W$ for x using the subset sampling algorithm of Mem. Compute $\pi = f_i(x)$ using the public evaluation algorithm for P on inputs s, x and w. Compute $K = H(\pi)$ and $\chi = \mathsf{SKE.Enc}(1^\lambda, K, m)$. The ciphertext is (x, χ).

Decryption. To decrypt a ciphertext (x, χ), compute $\pi = f_i(x)$ using the private evaluation algorithm for P on inputs i and x. Then decrypt χ under K using $\mathsf{SKE.Dec}$, and outputs the resulting decryption z. (z may be reject.)

8.2 Security

Theorem 2. In the above construction, suppose that Mem is hard, and the associated projective hash family $\mathsf{Project} = (F, I, X, L, \Pi, S, \alpha)$ is strongly universal$_2$ for each instance $\Lambda[X, L, W, R]$ of Mem. Moreover, suppose that the one-time symmetric-key encryption scheme SKE is secure in the sense of IND-CCA and it is ϵ-rejection secure for negligible ϵ. Then the proposed hybrid encryption scheme is secure in the sense of IND-CCA.

A proof is a generalization of that of Theorem 1. Roughly speaking, in the proof, if the challenge ciphertext is based upon application of the projective universal hash function f_i to an element $x^* \in L$, then the attack works as in the real case.

If $x^* \notin L$, then the following happens: At the beginning of the CCA attack, $\pi^* = f_i(x^*)$ (which is used as the symmetric key by $K^* = H(\pi^*)$) is totally uniform and secret from the point of view of the adversary. This is due to the strongly universal$_2$ property of the projective hash family Project. This information theoretic property of the symmetric key K^* remains as the attack progresses due to the fact that invalid queries are not decrypted due to the ϵ-rejection property of the SKE, where a ciphertext $C = (x, \chi)$ is invalid if $x \notin L$.

8.3 Examples

From [4, 5]. Let G be an Abelian group of order Q, where Q is a large prime. Let $X = G^2, W = Z_Q, L = \{(g_0^r, g_1^r) \mid r \in Z_Q\}$, where g_0, g_1 are two distinct generators of G. Then it is clear that the related membership problem Mem is hard if and only if the DDH assumption holds.

Let $\Gamma : G^2 \to Z_Q^n$ be an injective function for some n. Let $\Pi = S = G$ and $I = Z_Q^{2(n+1)}$. Define

$$\alpha(i_0, i_1, \cdots, i_n, j_0, j_1, \cdots, j_n) = (s_0, s_1, \cdots, s_n),$$

where $s_u = g_0^{i_u} g_1^{j_u}$ for $0 \le u \le n$. For $(x_0, x_1) \in X$, let $\Gamma(x_0, x_1) = (a_0, \cdots, a_n)$ and define

$$f_{(i_0,i_1,\cdots,i_n,j_0,j_1,\cdots,j_n)}(x_0, x_1) = x_0^{i_0+a_1 i_1 \cdots + a_n i_n} x_1^{j_0+a_1 j_1 \cdots + a_n j_n}.$$

(1) $\mathsf{Project} = (F, I, X, L, \Pi, S, \alpha)$ is a projective hash family because if $(x_0, x_1) = (g_0^r, g_1^r)$, then

$$\pi = f_{(i_0,i_1,\cdots,i_n,j_0,j_1\cdots,j_n)}(x_0, x_1) = (s_0 s_1^{a_1} \cdots s_n^{a_n})^r. \tag{10}$$

(2) Consider the probability space defined by choosing $(i_0, i_1, \cdots, i_n, j_0, j_1, \cdots, j_n) \in Z_Q^{2(n+1)}$ at random. For the example of [4, 5] we now have:

- For any $(x_0, x_1) \in X \setminus L$, $f_{(i_0,\cdots,i_n,j_0,\cdots,j_n)}(x_0, x_1)$ is uniformly distributed over G conditioned on fixed values of $(s_0, s_1, \cdots, s_n)$.
- For any $(x_0, x_1), (x_0^*, x_1^*) \in X \setminus L$ with $(x_0, x_1) \neq (x_0^*, x_1^*)$, we easily see that: $f_{(i_0,\cdots,i_n,j_0,\cdots,j_n)}(x_0, x_1)$ is uniformly distributed over G conditioned on fixed values of $(s_0, s_1, \cdots, s_n)$ and $\pi^* = f_{(i_0,\cdots,i_n,j_0,\cdots,j_n)}(x_0^*, x_1^*)$.

Hence $\mathsf{Project}$ is strongly universal$_2$.

Now from Sec.8.1, a concrete hybrid encryption scheme is obtained such that the ciphertext is $(g_0^r, g_1^r, \mathsf{SKE.Enc}(1^\lambda, K, m))$, where $K = H(\pi)$ and π is given by eq.(10). From Theorem 2, it is secure in the sense of IND-CCA if SKE satisfies the condition of the theorem. (This scheme is a TCR-free variant of Sec.4.3.)

Similarly, we can obtain efficient hybrid encryption schemes which are secure in the sense of IND-CCA under Quadratic Residuosity assumption and Paillier's Decision Composite Residuosity assumption, respectively.

Acknowledgment

We would like to thank Ronald Cramer and the anonymous reviewers for their helpful comments. The second author was supported by JPSP fellowship for research in Japan.

References

1. M. Bellare and P. Rogaway: Random Oracles are Practical: A Paradigm for Designing Efficient Protocols. ACM Conference on Computer and Communications Security 1993: 62-73
2. D.Boneh, Simplified OAEP for the RSA and Rabin Functions. CRYPTO 2001, pp.275-291 (2001)
3. R. Cramer and V. Shoup, "A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack," *Advances in Cryptology - CRYPTO'98*, Lecture Notes in Computer Science Vol. 1462, H. Krawczyk ed., Springer-Verlag, 1998.
4. R. Cramer and V. Shoup, "Universal hash proofs and a paradigm for chosen ciphertext secure public key encryption, against adaptive chosen ciphertext attack," *Advances in Cryptology - Eurocrypt'02*, Lecture Notes in Computer Science Vol. 2332, Springer-Verlag, 2002.
5. Full length version of [4]. http://shoup.net/papers/uhp.pdf
6. R. Cramer and V. Shoup, Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack, SIAM Journal on Computing, Volume 33, Number 1, pp. 167-226 (2003)
7. E. Fujisaki and T. Okamoto, Secure Integration of Asymmetric and Symmetric Encryption Schemes, Advances in Cryptology - CRYPTO '99, Lecture Notes in Computer Science Vol. 1666, M. Wiener ed., Springer-Verlag, 1999.
8. E.Fujisaki, T.Okamoto, D.Pointcheval, J.Stern, RSA-OAEP Is Secure under the RSA Assumption. CRYPTO 2001, pp.260-274 (2001)
9. H.Krawczyk, LFSR-based Hashing and Authentication, CRYPTO 1994, pp.129-139 (1994)
10. M.Naor and M.Yung, Universal One-Way Hash Functions and their Cryptographic Applications, STOC 1989, pp.33-43 (1989)
11. J.Rompel: One-Way Functions are Necessary and Sufficient for Secure Signatures, STOC 1990, pp.387-394 (1990)
12. V. Shoup, Using Hash Functions as a Hedge against Chosen Ciphertext Attack, EUROCRYPT 2000, pp.275-288 (2000)
13. V. Shoup, OAEP Reconsidered, CRYPTO 2001, pp.239-259 (2001)