Efficient Constructions of Deterministic Encryption from Hybrid Encryption and Code-Based PKE

Yang Cui[1,2], Kirill Morozov[1], Kazukuni Kobara[1,2], and Hideki Imai[1,2]

[1] Research Center for Information Security (RCIS),
National Institute of Advanced Industrial Science & Technology (AIST), Japan.
{y-cui, kirill.morozov, k-kobara, h-imai}@aist.go.jp
http://www.rcis.aist.go.jp/
[2] Chuo University, Japan.

Abstract. We build on the new security notion for deterministic encryption (PRIV) and the PRIV-secure schemes presented by Bellare et al at Crypto'07. Our work introduces: 1) A generic and efficient construction of deterministic length-preserving hybrid encryption, which is an improvement on the scheme sketched in the above paper; to our best knowledge, this is the first example of length-preserving hybrid encryption; 2) postquantum deterministic encryption (using the IND-CPA variant of code-based McEliece PKE) which enjoys a simplified construction, where the public key is re-used as a hash function.

Key words: deterministic encryption, hybrid encryption, code-based encryption, searchable encryption, database security

1 Introduction

Background. The notion of security against *privacy adversary* (denoted as PRIV) for deterministic encryption was pioneered by Bellare et al [2] featuring an upgrade from the standard onewayness property. Instead of not leaking the whole plaintext, the ciphertext was demanded to leak, roughly speaking, no more than the plaintext statistics does. In other words, the PRIV-security definition (formulated in a manner similar to the semantic security definition of [7]) requires that a ciphertext must be essentially useless for adversary who is to compute some predicate on the corresponding plaintext. Achieving PRIV-security demands two important assumptions: 1) the plaintext space must be large enough and have a smooth (i.e. high min-entropy) distribution; 2) the plaintext and the predicate are independent of the public key.

Constructions satisfying two flavors of PRIV-security are presented in [2]: against chosen-plaintext (CPA) and chosen-ciphertext (CCA) attacks. The following three PRIV-CPA constructions are introduced in the random oracle (RO) model. The generic Encrypt-with-Hash (EwH) primitive features replacing of

the coins used by the randomized encryption scheme with a hash of the public key concatenated with the message. The RSA deterministic OAEP (RSA-DOAEP) scheme provides us with length-preserving deterministic encryption. In the generic Encrypt-and-Hash (EaH) primitive, a "tag" in the form of the plaintext's hash is attached to the ciphertext of a randomized encryption scheme.

These results were extended by Boldyreva et al [4] and Bellare et al [3] presenting new extended definitions, proving relations between them, and introducing, among others, new constructions without random oracles.

Applications. The original motivation for this research comes from the demand on efficiently searchable encryption (ESE) in the database applications. Length-preserving schemes can also be used for encryption of legacy code and in the bandwidth-limited systems. Some more applications (although irrelevant to our work) to improving randomized encryption schemes were studied in [4, Sec. 8].

Motivation. The work [2, Sec. 5] sketches a method for encrypting long messages, but it is less efficient compared to the standard hybrid encryption, besides it is conjectured not to be length-preserving. Also, possible emerging of quantum computers raises demands for postquantum deterministic encryption schemes.

Our Contribution. In the random oracle model, we present a generic and efficient construction of length-preserving deterministic hybrid encryption. In a nutshell, we prove that the session key can be computed by concatenating the public key with the first message block and inputting the result into key derivation function. This is a kind of re-using the (sufficient) entropy of message, and it is secure due to the assumption that the message is high-entropy and independent of the key. Meanwhile, Bellare et al. employ the hybrid encryption in a conventional way, which first encrypts a random session key to further encrypt the data, obviously losing the length-preserving property. Hence, we show that the claim of Bellare et al [2, Sec. 5]: "However, if using hybrid encryption, RSA-DOAEP would no longer be length-preserving (since an encrypted symmetric key would need to be included with the ciphertext)" is overly pessimistic. To our best knowledge, this is the first example of length-preserving hybrid encryption.

For achieving postquantum deterministic encryption, we propose to plug in an IND-CPA secure variant [10] of the coding theory based (or code-based) McEliece PKE [9] into the generic constructions EaH and EwH, presented in [2, Sec. 5]. The McEliece PKE is believed to be resistant to quantum attacks, besides it has very fast encryption algorithm. Moreover, we point out a significant simplification: the public key (which is a generating matrix of some linear code) can be re-used as hash function.

Related Work. The deterministic hybrid encryption scheme is based on the same principle as the RSA-DOAEP scheme of [2, Sec. 5], we just fill the gap which was overlooked there.

Organization. The paper will be organized in the following way: Sec.2 provides the security definitions of deterministic encryption. Sec.3 gives the proposed generic and efficient construction of deterministic hybrid encryption, which leads to the first length-preserving construction, immediately. In Sec.4, we will provide

deterministic encryption from the code-based PKE, which is postquantum secure and efficient due to the good property of the underlying PKE scheme. Next, in Sec.5, we further discuss how to extend the PRIV security to the chosen-ciphertext attack (CCA) scenario.

2 Preliminaries

Denote by "$|x|$" the cardinality of x. Denote by $\hat{\boldsymbol{x}}$ the vector and by $\hat{\boldsymbol{x}}[i]$ the i-th component of $\hat{\boldsymbol{x}}$ $(1 \leq i \leq |\hat{\boldsymbol{x}}|)$. Write $\hat{\boldsymbol{x}}||\hat{\boldsymbol{y}}$ for concatenation of vectors $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$. Let $x \leftarrow_R X$ denote the operation of picking x from the set X uniformly at random. Denote by $z \leftarrow A(x, y, ...)$ the operation of running algorithm A with input $(x, y, ...)$, to output z. Write $\log x$ as the logarithm with base 2. We also write $\Pr[A(x) = y : x \leftarrow_R X]$ the probability that A outputs y corresponding to input x, which is sampled from X. We say a function $\epsilon(k)$ is negligible, if for any constant c, there exists $k_0 \in \mathbb{N}$, such that $\epsilon < (1/k)^c$ for any $k > k_0$.

A public key encryption (PKE) scheme Π consists of a triple of algorithms $(\mathcal{K}, \mathcal{E}, \mathcal{D})$. The key generation algorithm $\mathcal{K}$ outputs a pair of public and secret keys $(\mathsf{pk}, \mathsf{sk})$ taking on input 1^k, a security parameter k in unitary notation. The encryption algorithm $\mathcal{E}$ on input pk and a plaintext $\hat{\boldsymbol{x}}$ outputs a ciphertext c. The decryption algorithm $\mathcal{D}$ takes sk and c as input and outputs the plaintext message $\hat{\boldsymbol{x}}$. We require that for any key pair $(\mathsf{pk}, \mathsf{sk})$ obtained from $\mathcal{K}$, and any plaintext $\hat{\boldsymbol{x}}$ from the plaintext space of Π, $\hat{\boldsymbol{x}} \leftarrow \mathcal{D}(\mathsf{sk}, \mathcal{E}(\mathsf{pk}, \hat{\boldsymbol{x}}))$.

Definition 1 (PRIV [2]). *Let a probabilistic polynomial-time (PPT) adversary $\mathcal{A}_{DE}$ against the privacy of the deterministic encryption $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, be a pair of algorithms $\mathcal{A}_{DE} = (\mathcal{A}_f, \mathcal{A}_g)$, where $\mathcal{A}_f, \mathcal{A}_g$ do not share any random coins or state. The advantage of adversary is defined as follows,*

$$\mathbf{Adv}^{priv}_{\Pi,\mathcal{A}_{DE}}(k) = \Pr[\mathbf{Exp}^{priv-1}_{\Pi,\mathcal{A}_{DE}}(k) = 1] - \Pr[\mathbf{Exp}^{priv-0}_{\Pi,\mathcal{A}_{DE}}(k) = 1]$$

where experiments are described as:

Experiment $\mathbf{Exp}^{priv-1}_{\Pi,\mathcal{A}_{DE}}(k)$:	**Experiment** $\mathbf{Exp}^{priv-0}_{\Pi,\mathcal{A}_{DE}}(k)$:
$(\mathsf{pk}, \mathsf{sk}) \leftarrow_R \mathcal{K}(1^k)$,	$(\mathsf{pk}, \mathsf{sk}) \leftarrow_R \mathcal{K}(1^k)$,
$(\hat{\boldsymbol{x}}_1, t_1) \leftarrow_R \mathcal{A}_f(1^k)$,	$(\hat{\boldsymbol{x}}_0, t_0) \leftarrow_R \mathcal{A}_f(1^k), (\hat{\boldsymbol{x}}_1, t_1) \leftarrow_R \mathcal{A}_f(1^k)$,
$c \leftarrow_R \mathcal{E}(1^k, \mathsf{pk}, \hat{\boldsymbol{x}}_1)$,	$c \leftarrow_R \mathcal{E}(1^k, \mathsf{pk}, \hat{\boldsymbol{x}}_0)$,
$g \leftarrow_R \mathcal{A}_g(1^k, \mathsf{pk}, c)$;	$g \leftarrow_R \mathcal{A}_g(1^k, \mathsf{pk}, c)$;
return 1 if $g = t_1$, *else return 0*	*return 1 if* $g = t_1$, *else return 0*

We say that Π is PRIV secure*, if $\mathbf{Adv}^{priv}_{\Pi,\mathcal{A}_{DE}}(k)$ is negligible, for any PPT $\mathcal{A}_{DE}$ with high min-entropy, where $\mathcal{A}_{DE}$ has a high min-entropy $\mu(k)$ means that $\mu(k) \in \omega(log(k))$, and $\Pr[\hat{\boldsymbol{x}}[i] = x : (\hat{\boldsymbol{x}}, t) \leftarrow_R \mathcal{A}_m(1^k)] \leq 2^{-\mu(k)}$ for all k, all $1 \leq i \leq |\hat{\boldsymbol{x}}|$, and any $x \in \{0,1\}^*$.*

In the underlying definition, the advantage of privacy adversary could be also written as

$$\mathbf{Adv}^{priv}_{\Pi,\mathcal{A}_{DE}}(k) = 2\Pr[\mathbf{Exp}^{priv-b}_{\Pi,\mathcal{A}_{DE}}(k) = b] - 1$$

where $b \in \{0,1\}$ and probability is taken over the choice of all of the random coins in the experiments.
Remarks. 1) The encryption algorithm Π need not be deterministic per se. For example, in a randomized encryption scheme, the random coins can be fixed in an appropriate way to yield a deterministic scheme (as explained in Sec.4);
2) As argued in [2], $\mathcal{A}_f$ has no access to the pk and $\mathcal{A}_g$ does not know the chosen plaintext input to encryption oracle by $\mathcal{A}_f$. This is required because the public key itself carries some non-trivial information about the plaintext if the encryption is deterministic.[3] Thus, equipping either $\mathcal{A}_f$ or $\mathcal{A}_g$ with both the public key and free choice of an input plaintext in the way of conventional indistinguishability notion [7] of PKE, the PRIV security cannot be achieved.

It is possible to build PRIV security from indistinguishability (IND) security, as observed in [2]. In the following, we recall the notion of IND security.

Definition 2 (IND-CPA). *We say a scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is IND-CPA secure, if the advantage $\mathbf{Adv}^{ind}_{\Pi,\mathcal{A}}$ of any PPT adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ is negligible, (let s be the state information of $\mathcal{A}_1$, and $\hat{b} \in \{0,1\}$):*

$$\mathbf{Adv}^{ind}_{\Pi,\mathcal{A}}(k) = 2 \cdot \Pr\left[\begin{array}{l} \hat{b} = b : (\mathsf{pk}, \mathsf{sk}) \leftarrow_R \mathcal{K}(1^k), \\ (x_0, x_1, s) \leftarrow_R \mathcal{A}_1(1^k, \mathsf{pk}), \\ b \leftarrow_R \{0,1\}, c \leftarrow_R \mathcal{E}(1^k, \mathsf{pk}, x_b), \\ \hat{b} \leftarrow_R \mathcal{A}_2(1^k, c, s) \end{array}\right] - 1$$

Remark. IND security is required by a variety of cryptographic primitives. However, for an efficiently searchable encryption used in database applications, IND secure encryption may be considered as overkill. For such a strong encryption, it is not known how to arrange fast (i.e. logarithmic in the database size) search.

IND secure symmetric key encryption (SKE) has been carefully discussed in the literature, such as [6, Sec.7.2]. Given a key $K \in \{0,1\}^k$ and message m, an encryption algorithm outputs a ciphertext χ. Provided χ and K, a decryption algorithm outputs the message m uniquely. Note that for a secure SKE, outputs of the encryption algorithm could be considered uniformly distributed in the range, when encrypted under independent session keys. Besides, it is easy to build IND secure SKE.

Definition 3 (IND-CPA SKE). *A symmetric key encryption (SKE) scheme $\Lambda = (\mathcal{K}_{SK}, \mathcal{E}_{SK}, \mathcal{D}_{SK})$ with key space $\{0,1\}^k$, is indistinguishable against chosen plaintext attack (IND-CPA) if the advantage of any PPT adversary $\mathcal{B}$, $\mathbf{Adv}^{ind-cpa}_{\Lambda,\mathcal{B}}$ is negligible, where*

$$\mathbf{Adv}^{ind-cpa}_{\Lambda,\mathcal{B}}(k) = 2 \cdot \Pr\left[\begin{array}{l} \hat{b} = b : K \leftarrow_R \{0,1\}^k, b \leftarrow_R \{0,1\}, \\ \hat{b} \leftarrow_R \mathcal{B}^{\mathsf{LOR}(K,\cdot,\cdot,b)}(1^k) \end{array}\right] - 1,$$

[3] In other words, suppose that in Def. 1, $\mathcal{A}_f$ knows pk. Then, $\mathcal{A}_f$ can assign t_1 to be the ciphertext c, and hence $\mathcal{A}_g$ always wins the game (returns 1). Put it differently, although $\mathcal{A}_f$ and $\mathcal{A}_g$ are not allowed to share a state, knowledge of pk can help them to share it anyway.

where a left-or-right oracle $\mathsf{LOR}(K, M_0, M_1, b)$ *returns* $\chi \leftarrow_R \mathcal{E}_{SK}(K, M_b)$. *Adversary* $\mathcal{B}$ *is allowed to ask* LOR *oracle, with two chosen message* M_0, M_1 *(*$M_0 \neq M_1$, $|M_0| = |M_1|$*).*

Hybrid Encryption. In the seminal paper by Cramer and Shoup [6], the idea of hybrid encryption is rigorously studied. Note that typically, PKE is applied in key distribution process due to its expensive computational cost, while SKE is typically used for encrypting massive data flow using a freshly generated key for each new session. In hybrid encryption, PKE and SKE work in tandem: a randomly generated session key is first encrypted by PKE, then the plaintext is further encrypted on the session key by SKE. Hybrid encryption is more commonly used in practice than a sole PKE, since encryption/decryption of the former is substantially faster for long messages.

McEliece PKE. (denoted Π_M) Consists of the following triple of algorithms $(\mathcal{K}_M, \mathcal{E}_M, \mathcal{D}_M)$.

1. Key generation $\mathcal{K}_M$: On input λ, output $(\mathsf{pk}, \mathsf{sk})$. $n, t \in \mathbb{N}$, $t \ll n$
   - sk (Private Key): (S, φ, P)
     G': $l \times n$ generating matrix of a binary irreducible $[n, l]$ Goppa code which can correct a maximum of t errors. φ is an efficient bounded distance decoding algorithm of the underlying code, S: $l \times l$ non-singular matrix, P: $n \times n$ permutation matrix, chosen at random.
   - pk (Public Key): (G, t)
     G: $l \times n$ matrix given by a product of three matrices $SG'P$.
2. Encryption $\mathcal{E}_M$: Given pk and an l-bit plaintext m, randomly generate n-bit e with Hamming weight t, output ciphertext $c = mG \oplus e$.
3. Decryption $\mathcal{D}_M$: On input c, output m with private key sk.
   - Compute $cP^{-1} = (mS)G' \oplus eP^{-1}$, where P^{-1} is an inverse matrix of P.
   - Error correcting algorithm φ corresponding to G' applies to compute $mS = \varphi(cP^{-1})$.
   - Compute the plaintext $m = (mS)S^{-1}$.

IND-CPA security of the McEliece PKE can be achieved by padding the plaintext with a random bit-string r, $|r| = \lceil a \cdot l \rceil$ for some $0 < a < 1$. We refer to [10] for details.

3 Secure Deterministic Hybrid Encryption

In this section, we will present a generic composition of PKE and SKE to obtain deterministic hybrid encryption. Interestingly, the situation is different from conventional hybrid encryption. In that case, the overhead of communication cost includes at least the size of the session key, even if we pick the PKE scheme being a (length-preserving) one-way trapdoor permutation, e.g. RSA.

However, we notice that in PRIV security definition, both of public key and plaintext are not simultaneously known by $\mathcal{A}_f$ or $\mathcal{A}_g$. Hence, one can save on generating and encrypting a random session key. Instead, the secret session key

could be extracted from the combination of public key and plaintext which are available to a legal user contrary to the adversary. As we show next, such an approach may need a little higher min-entropy, but it works in principle.

3.1 Generic Composition of PRIV-secure PKE and IND-CPA Symmetric Key Encryption

Given a PRIV secure PKE scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, and an IND-CPA secure SKE scheme $\Lambda = (\mathcal{K}_{SK}, \mathcal{E}_{SK}, \mathcal{D}_{SK})$, we can achieve a deterministic hybrid encryption $HE = (\mathcal{K}_H, \mathcal{E}_H, \mathcal{D}_H)$. In the following, $H : \{0,1\}^* \mapsto \{0,1\}^k$ is a key derivation function (KDF), modeled as a random oracle. In the following section, we simply write input vector $\hat{\boldsymbol{x}}$ as x with length of $|\hat{\boldsymbol{x}}| = v$. Wlog, parse $x = \bar{x}||\underline{x}$, where the $|\bar{x}|$ and $|\underline{x}|$ are equivalent to the input domain of Π and Λ, respectively.

$\mathcal{K}_H(1^k)$:	$\mathcal{E}_H(\mathsf{pk}, x)$:	$\mathcal{D}_H(\mathsf{sk}, c)$:
$(\mathsf{pk}, \mathsf{sk}) \leftarrow_R \mathcal{K}(1^k)$	Parses x to $\bar{x}\|\|\underline{x}$	Parse c to $\psi\|\|\chi$
Return $(\mathsf{pk}, \mathsf{sk})$	$\psi \leftarrow_R \mathcal{E}(1^k, \mathsf{pk}, \bar{x})$	$\bar{x} \leftarrow \mathcal{D}(\mathsf{sk}, \psi)$
	$K \leftarrow H(\mathsf{pk}\|\|\bar{x})$	$K \leftarrow H(\mathsf{pk}\|\|\bar{x})$
	$\chi \leftarrow_R \mathcal{E}_{SK}(K, \underline{x})$	$\underline{x} \leftarrow \mathcal{D}_{SK}(K, \chi)$
	Return $c = \psi\|\|\chi$	Return $x = \bar{x}\|\|\underline{x}$

Table 1. Generic Construction of Deterministic Hybrid Encryption

In the Table 1, the proposed construction is simple, efficient, and can be generically built from any PRIV PKE and IND-CPA SKE. Note that the secret session key is required to have high min-entropy in order to deny a brute-force attack to SKE. However, thanks to the PRIV security, the high min-entropy requirement is inherently fulfilled for any PPT privacy adversary, so that we can build a reduction of security of the deterministic hybrid encryption to security of deterministic PKE. Next, we will provide a sketch of our proof.

3.2 Security Proof

Theorem 1. *In the random oracle model, given a PRIV PKE scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, and an IND-CPA SKE scheme $\Lambda = (\mathcal{K}_{SK}, \mathcal{E}_{SK}, \mathcal{D}_{SK})$, if there is a PRIV adversary $\mathcal{A}_H$ against the hybrid encryption $HE = (\mathcal{K}_H, \mathcal{E}_H, \mathcal{D}_H)$, then there exists PRIV adversary $\mathcal{A}$ or IND-CPA adversary $\mathcal{B}$, s.t.*

$$\mathbf{Adv}^{priv}_{HE,\mathcal{A}_H}(k) \leq \mathbf{Adv}^{priv}_{\Pi,\mathcal{A}}(k) + \mathbf{Adv}^{ind-cpa}_{\Lambda,\mathcal{B}}(k) + q_h v / 2^{\mu}$$

where q_h is an upper bound on the number of queries to random oracle H, v is the plaintext size of Π, μ is defined by high min-entropy of PRIV security of Π.

Proof. Since we assume a PPT adversary $\mathcal{A}_H = (\mathcal{A}_f, \mathcal{A}_g)$ against the HE scheme, according to the definition of PRIV, there must be a non-negligible advantage in the following experiments.

Experiment $\mathbf{Exp}^{priv-1}_{HE,\mathcal{A}_H}(k)$:
$(\mathsf{pk},\mathsf{sk}) \leftarrow_R \mathcal{K}(1^k)$;
$(x_1,t_1) \leftarrow_R \mathcal{A}_f(1^k)$;
Parse x_1 to $\bar{x}_1||\underline{x}_1$;
$\psi \leftarrow_R \mathcal{E}(1^k,\mathsf{pk},\bar{x}_1)$;
$K \leftarrow H(\mathsf{pk}||\bar{x}_1)$;
$\chi \leftarrow_R \mathcal{E}_{SK}(K,\underline{x}_1)$;
$c \leftarrow \psi||\chi$;
$g \leftarrow_R \mathcal{A}_g(1^k,\mathsf{pk},c)$;
return 1 if $g = t_1$,
else return 0

Experiment $\mathbf{Exp}^{priv-0}_{HE,\mathcal{A}_H}(k)$:
$(\mathsf{pk},\mathsf{sk}) \leftarrow_R \mathcal{K}(1^k)$;
$(x_0,t_0) \leftarrow_R \mathcal{A}_f(1^k)$, $(x_1,t_1) \leftarrow_R \mathcal{A}_f(1^k)$;
Parse x_0 to $\bar{x}_0||\underline{x}_0$;
$\psi' \leftarrow_R \mathcal{E}(1^k,\mathsf{pk},\bar{x}_0)$;
$K' \leftarrow H(\mathsf{pk}||\bar{x}_0)$;
$\chi' \leftarrow_R \mathcal{E}_{SK}(K',\underline{x}_0)$;
$c' \leftarrow \psi'||\chi'$;
$g \leftarrow_R \mathcal{A}_g(1^k,\mathsf{pk},c')$;
return 1 if $g = t_1$,
else return 0

More precisely, if a successful adversary exists, then

$$\mathbf{Adv}^{priv}_{HE,\mathcal{A}_H}(k) = \Pr[\mathbf{Exp}^{priv-1}_{HE,\mathcal{A}_H}(k) = 1] - \Pr[\mathbf{Exp}^{priv-0}_{HE,\mathcal{A}_H}(k) = 1]$$

is non-negligible for some $\mathcal{A}_H$. Next we present a simulator which gradually modifies the above experiments such that the adversary does not notice it. Our goal is to show that $\mathbf{Adv}^{priv}_{HE,\mathcal{A}_H}(k)$ is almost as big as the corresponding advantages defined for PRIV security of the PKE scheme and IND-CPA security of the SKE scheme, which are assumed negligible.

Because of the high min-entropy requirement of PRIV adversary, it is easy to see that $x_0 \neq x_1$, except with negligible probability. Thus, there must be $\bar{x}_0 \neq \bar{x}_1$ or $\underline{x}_0 \neq \underline{x}_1$, or both. Hence, we need to consider the following cases.

Case [$\bar{x}_0 \neq \bar{x}_1$] Since $x_0 \neq x_1$ and $\bar{x}_0 \neq \bar{x}_1$, the right part of x_b ($b \in \{0,1\}$), could be equal or not.

- When $\underline{x}_0 = \underline{x}_1$, the adversary has two targets, such as Π and Λ in two experiments. First look at the SKE scheme Λ. In this case, the inputs to Λ in two experiments are the same, but still unknown to $\mathcal{A}_g$. The key derivation function H outputs $K \leftarrow H(\mathsf{pk}||\bar{x}_1)$ and $K' \leftarrow H(\mathsf{pk}||\bar{x}_0)$. Since $\bar{x}_0 \neq \bar{x}_1$, we have $K \neq K'$. Note that $\mathcal{A}_g$ does not know x_0 nor x_1, thus does not know K, K', either. Then, $\mathcal{A}_g$ must tell which of χ, χ' is the corresponding encryption under the unknown keys without knowing $\underline{x}_0, \underline{x}_1 (\underline{x}_0 = \underline{x}_1)$, which is harder than breaking IND-CPA security and that could be bounded by $\mathbf{Adv}^{ind-cpa}_{\Lambda,\mathcal{B}}(k)$.
  On the other hand, the adversary can also challenge the PKE scheme Π to distinguish two experiments, but it will break the PRIV security. More precisely, the advantage in distinguishing ψ, ψ' with certain K, K' is at most $\mathbf{Adv}^{priv}_{\Pi,\mathcal{A}}(k)$, since K, K' are not output explicitly and unavailable to adversary.
- when $\underline{x}_0 \neq \underline{x}_1$, this case is similar to the above, except that the inputs to Λ are different. $\mathcal{A}_g$ can do nothing given χ, χ' only, hence $\mathcal{A}_g$'s possible attack must be focused on Π, and its advantage can be bounded by $\mathbf{Adv}^{priv}_{\Pi,\mathcal{A}}(k)$.

Case [$\underline{x}_0 \neq \underline{x}_1$] Similarly, there must be either $\bar{x}_0 \neq \bar{x}_1$ or $\bar{x}_0 = \bar{x}_1$.

- when $\bar{x}_0 = \bar{x}_1$, the same session key $K \leftarrow H(\mathsf{pk}||\bar{x}_b)$ ($b \in \{0,1\}$) is used for Λ. In this case, the ciphertexts ψ, ψ' are the same, adversary will focus on distinguishing the χ, χ'. Note that $\mathcal{A}_f$ cannot compute K even though he knows the $\bar{x}_0 = \bar{x}_1$, because pk is not known to him (otherwise, it will break the PRIV security of Π immediately!). Thus, the successful distinguishing requires $\mathcal{A}_g$ to choose the same $\bar{x}_0 = \bar{x}_1$ when querying to the random oracle. Then, $\mathcal{A}_g$ has a harder game than IND-CPA (because it does not know $\underline{x}_0, \underline{x}_1$), whose advantage is bounded by $\mathbf{Adv}^{ind-cpa}_{\Lambda,\mathcal{B}}(k)$. In order to be sure that adversary $(\mathcal{A}_f, \mathcal{A}_g)$ mounting a brute-force attack to find out the session key of Λ cannot succeed, the probability to find the key in searching all the random oracle queries should be taken into account as well. Suppose that adversary makes at most q_h queries to its random oracle, and the Π's plaintext size is v. Then, this probability could be upper bounded by $q_h v/2^{\mu}$ (Note that this bound is in nature similar to that in [2, Sec.6.1]).
- when $\bar{x}_0 \neq \bar{x}_1$, as we have discussed above, this will break the PRIV security of Π, and advantage of adversary could be bounded by $\mathbf{Adv}^{priv}_{\Pi,\mathcal{A}}(k)$.

Summarizing, we conclude that in all cases when $(\mathcal{A}_f, \mathcal{A}_g)$ intends to break the PRIV security of our HE scheme, its advantage of distinguishing two experiments is bounded by the sum of $\mathbf{Adv}^{priv}_{\Pi,\mathcal{A}}(k)$, $q_h v/2^{\mu}$ and $\mathbf{Adv}^{ind-cpa}_{\Lambda,\mathcal{B}}(k)$. □

Length-preserving Deterministic Hybrid Encryption.

The first length-preserving PRIV PKE scheme is RSA-DOAEP due to [2]. The length-preserving property is important in practical use, such as bandwidth-restricted applications. RSA-DOAEP makes use of the RSA trapdoor permutation and with a modified 3-round Feistel network achieves the same sizes of input and output. As we have proved in Theorem 1, a construction proposed in Table 1 leads to a deterministic hybrid encryption.

In particular, RSA-DOAEP + IND-CPA SKE ⇒ a length-preserving deterministic hybrid encryption, because both RSA-DOAEP and IND-CPA SKE are length-preserving. Note that in [2, Sec.5.2], it is argued that RSA-DOAEP based hybrid encryption scheme cannot be length-preserving any more, because a random session key has to be embedded in RSA-DOAEP. However, by re-using the knowledge of public key pk and a part of the message, we can indeed build the first length-preserving deterministic hybrid encryption, which is not only convenient in practice, but also meaningful in theory.

4 Deterministic Encryption from Code-based PKE

From a postquantum point of view, it is desirable to obtain deterministic encryption based on assumptions other than RSA or discrete log. Code-based PKE, such as McEliece PKE [9] is considered a promising candidate after being carefully studied for over thirty years.

To our surprise, it is not the only motivation to achieve deterministic encryption from code-based PKE. Another good property of the McEliece PKE and

its variants is that its public key could be used as a hash function to digest the message, which is originally noted in Stern's paper [11], and recently designed by [1, 8]. The advantage that public key itself is able to work as a hash function, can do us a favor to build efficient postquantum deterministic encryption. We call this Hidden Hash (HH) property of McEliece PKE. Henceforth, we assume that this function behaves as a random oracle.

In [2], two constructions satisfying PRIV security have been proposed: Encrypt-with-Hash (EwH) and Encrypt-and-Hash (EaH). Adapting the HH property of the McEliece PKE to the both constructions, we can achieve PRIV secure deterministic encryption. For proving PRIV security, we require the McEliece PKE to be IND-CPA secure, which has been proposed in [10]. (The proofs are deferred to the full version of this paper).

Construction of EwH. Let $\Pi_M = (\mathcal{K}_M, \mathcal{E}_M, \mathcal{D}_M)$ be the IND-CPA McEliece PKE as described in Section 2, based on $[n, l, 2t+1]$ Goppa code family, with l_p-bit padding where $l_p = \lceil a \cdot l \rceil$ for some $0 < a < 1$, and plaintext length $l_m = l - l_p$. Let $\mathcal{H}$ be a hash family defined over a set of public keys of the McEliece PKE. $H_M : \{0,1\}^{l_m} \mapsto \{0,1\}^{l_p + \log \sum_{i=1}^{t} \binom{n}{t}}$ and $H_N : \{0,1\}^{l_m} \mapsto \{0,1\}^{2k}$ are uniquely defined by 1^k and pk. Without knowledge of pk, there is no way to compute H_M or H_N (refer to [1, 8] for details). e is an error vector, s.t. $|e| = n$ with Hamming weight $Hw(e) = t$. According to Cover's paper [5], it is quite efficient to find an injective mapping to encode the (short) bit string r_e into e, and vice versa.

Our EwH scheme is presented in Table 2.

$\mathcal{K}(1^k)$:	$\mathcal{E}(\mathsf{pk}, H_N, x)$:	$\mathcal{D}(\mathsf{sk}, H_M, c)$:
$(\mathsf{pk}, \mathsf{sk}) \leftarrow_R \mathcal{K}_M(1^k)$	$R \leftarrow H_M(x)$	$x, r', e \leftarrow \mathcal{D}_M(\mathsf{sk}, c)$
$H_M \leftarrow \mathcal{H}(1^k, \mathsf{pk})$	Parse R to $r \| r_e$	Decode e to r'_e
Return $(\mathsf{pk}, H_M, \mathsf{sk})$	Encode r_e to e	$R' \leftarrow r' \| r'_e$, $R \leftarrow H_M(x)$
	$c \leftarrow \mathcal{E}_M(\mathsf{pk}, r \| x; e)$	Return x if $R = R'$
	Return c	Otherwise, return $\perp$

Table 2. Construction of EwH Deterministic Encryption

$\mathcal{K}(1^k)$:	$\mathcal{E}(\mathsf{pk}, H_M, x)$:	$\mathcal{D}(\mathsf{sk}, H_M, c \| T)$:
$(\mathsf{pk}, \mathsf{sk}) \leftarrow_R \mathcal{K}_M(1^k)$	$T \leftarrow H_N(x)$	$x, r, e \leftarrow \mathcal{D}_M(\mathsf{sk}, c)$
$H_N \leftarrow \mathcal{H}(1^k, \mathsf{pk})$	$r \leftarrow_R \{0,1\}^{l_p}$	$T' \leftarrow H_N(x)$
Return $(\mathsf{pk}, H_M, \mathsf{sk})$	$e \leftarrow_R \{0,1\}^n$, s.t. $Hw(e) = t$	Return x if $T = T'$
	$c \leftarrow \mathcal{E}_M(\mathsf{pk}, r \| x; e)$	Otherwise, return $\perp$
	Return $c \| T$	

Table 3. Construction of EaH Deterministic Encryption

Note that compared with the EwH scheme proposed by Bellare et al. [2], our scheme does not need to include pk into the hash, because hash function H_M itself is made of pk. Public key pk could be considered as a part of the algorithm of the hash function, as well. When we model H_M as a random oracle, we can easily prove the PRIV security in a similar way as Bellare et al's EwH.

A more favorable, efficiently searchable encryption (ESE) with PRIV security is EaH. EaH aims to model the practical scenario in database security, where a deterministic encryption of some keywords works as a tag attached to the

encrypted data. To search the target data, it is only required to compute the deterministic tag and compare it within the database, achieving a search time which is logarithmic in database size.

Construction of EaH. The description of McEliece PKE is similar to the above. EaH scheme is described in Table 3. The HH property is employed in order to achieve PRIV secure efficiently searchable encryption.

5 Concluding Remarks

Extension to Chosen-Ciphertext Security. Above, we have proposed several PRIV secure deterministic encryption schemes, in CPA case. A stronger attack scenario, CCA, requires a little more care. As commented in [2], PRIV-CCA could be obtained from PRIV-CPA scheme with some additional cost, such as one-time signatures or other authentication techniques to deny a CCA attacker. We can employ those techniques to lift up CPA to CCA. The important issue is that we have achieved very efficient PRIV-CPA secure building blocks which enjoy some advantages over previous works.

Open Question. Proving our constructions secure in the standard model is an open question and the topic of our future work.

References

1. D. Augot, M. Finiasz and N. Sendrier, "A Family of Fast Syndrome Based Cryptographic Hash Functions," Mycrypt 2005, LNCS 3715, pp. 64-83, 2005.
2. M. Bellare, A. Boldyreva and A. O'Neill, "Deterministic and Efficiently Searchable Encryption," CRYPTO'07, LNCS 4622, pp. 535-552, 2007.
3. M. Bellare, M. Fischlin, A. O'Neill and T. Ristenpart, "Deterministic Encryption: Definitional Equivalences and Constructions without Random Oracles." CRYPTO'08, LNCS 5157, pp. 360-378, 2008.
4. A. Boldyreva, S. Fehr and A. O'Neill, "On Notions of Security for Deterministic Encryption, and Efficient Constructions without Random Oracles," CRYPTO'08, LNCS 5157, pp. 335-359, 2008.
5. T. Cover, "Enumerative source encoding," IEEE IT 19(1), pp. 73-77, 1973.
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. S. Goldwasser and S. Micali. Probabilistic encryption. Journal of Computer and System Sciences, 28(2), pp. 270-299, 1984.
8. M. Finiasz, "Syndrome Based Collision Resistant Hashing," PQCrypto 2008, LNCS 5299, pp. 137-147, 2008.
9. R. J. McEliece. "A public-key cryptosystem based on algebraic coding theory," Deep Space Network Progress Rep. 42-44, pp. 114-116, 1978.
10. R. Nojima, H. Imai, K. Kobara and K. Morozov, "Semantic Security for the McEliece Cryptosystem without Random Oracles," Designs, Codes and Cryptography, vol. 49, no. 1-3, pp. 289-305, 2008.
11. J. Stern. "A new identification scheme based on syndrome decoding," CRYPTO'93, LNCS 773, pp. 13-21, 1993.