A Closer Look at Anonymity and Robustness in Encryption Schemes

Payman Mohassel

Computer Science Department, University of Calgary
pmohasse@cpsc.ucalgary.ca

Abstract. In this work, we take a closer look at *anonymity* and *robustness* in encryption schemes. Roughly speaking, an anonymous encryption scheme hides the identity of the secret-key holder, while a robust encryption scheme guarantees that every ciphertext can only be decrypted to a valid plaintext under the intended recipient's secret key.
In case of anonymous encryption, we show that if an anonymous PKE or IBE scheme (in presence of CCA attacks) is used in a hybrid encryption, all bets regarding the anonymity of the resulting encryption are off. We show that this is the case even if the symmetric-key component is anonymous. On the positive side, however, we prove that if the key-encapsulation method is, additionally *weakly robust* the resulting hybrid encryption remains anonymous. Some of the existing anonymous encryption schemes are known to be weakly robust which makes them more desirable in practice.
In case of robust encryption, we design several *efficient* constructions for transforming any PKE/IBE scheme into weakly and strongly robust ones. Our constructions only add a minor computational overhead to the original schemes, while achieving better ciphertext sizes compared to the previous constructions. An important property of our transformations is that they are non-keyed and do not require any modifications to the public parameters of the original schemes.
We also introduce a relaxation of the notion of robustness we call *collision-freeness*. We primarily use collision-freeness as an *intermediate notion* by showing a more efficient construction for transforming any collision-free encryption scheme into a strongly robust one. We believe that this simple notion can be a plausible replacement for robustness in some scenarios in practice. The advantage is that most existing schemes seem to satisfy collision-freeness without any modifications.

1 Introduction

The classical definitions of security for encryption schemes are mainly concerned with the secrecy of encrypted data. Particularly, the widely accepted notions of indistinguishability and non-malleability under chosen plaintext and ciphertext attacks [15,19,12], are all directed at capturing various aspects of data-secrecy in encryption schemes. However, since encryption schemes are employed in a wide range of applications, one often requires them to satisfy additional properties.

Two such properties, which have been the subject of formal studies in the cryptographic literature, are *anonymity* [5,2] and *robustness* [3]. Anonymity helps keep the identity of the key-holders in an encryption scheme private, while robustness provides a layer of protection against misuse or error by ensuring that a single ciphertext can only be decrypted by the intended user. In this paper we study several aspects of anonymity and robustness in public-key and identity-based encryption schemes.

1.1 Anonymity of Hybrid Encryption Schemes

The concept of anonymity for encryption schemes has been around for sometime but was first formalized in the context of symmetric-key encryption [1,11,14] and was later extended to the case of public-key encryption (PKE) and identity-based encryption (IBE) [5,2]. Several PKE and IBE schemes in the literature such as the Cramer-Shoup [10], and the Boyen-Waters [9] in the standard model, and DHIES [4] and Boneh-Franklin [8] in the random oracle model are shown to be anonymous.

However, in most cases, PKE and IBE schemes are used as key encapsulation methods (KEM) to encrypt a random key which is then used by a symmetric-key data encapsulation method (DEM) to encrypt the message itself. It is well known that if the KEM component is IND-CCA secure and the DEM component is (one-time) IND-CCA, the resulting hybrid encryption is also IND-CCA secure (e.g. see [10]).[1] From a practical point of view, it is important to determine whether similar statements can be made when considering anonymity.

A NEGATIVE RESULT. At first glance, it seems that the symmetric-key component is harmless as far as anonymity is concerned since it only encrypts a message using a random secret key, which is unlikely to reveal additional information about the public key or the identity (this is in fact the case for CPA attacks). However, somewhat surprisingly, we show that this intuition is wrong in presence of *chosen ciphertext attacks*. Particularly, we show a counterexample by building an anonymous-CCA (ANON-CCA) secure PKE/IBE scheme and a symmetric-key IND-CCA encryption, where it is easy to break the anonymity of the resulting hybrid construction. The negative result extends to the case when the symmetric-key component is also anonymous. An important implication is that:

Designing ANON-CCA PKE or IBE schemes is not sufficient for providing anonymity in practice where, more often than not, encryption schemes are used in hybrid constructions.

A POSITIVE RESULT. On the positive side, we show that if one further assumes that the KEM component is *weakly-robust* (see Section 2 for the definition), the resulting hybrid encryption is in fact ANON-CCA. This implies that despite our

[1] Note that the KEM/DEM framework is more general than hybrid encryption but here we are focus on the KEM/DEM framework in the context of hybrid encryption schemes.

negative result, for most ANON-CCA schemes we know of such as the Boneh-Franklin IBE, the Cramer-Shoup PKE, and the DHIES PKE all of which are known to be weakly-robust [3] (in the appropriate model), using them as part of a hybrid construction preserves their anonymity. The same is however not true for the Boyen-Waters anonymous IBE scheme which is shown not to be weakly robust.

This result reemphasizes the close connection between anonymity and robustness and provides additional motivation to study the robustness property when designing anonymous encryption schemes.

1.2 Robustness

Informally speaking, *weak* robustness requires that a ciphertext does not decrypt to a valid plaintext under distinct secret keys for two different identities. A *stronger* version of robustness requires this to be the case even for adversarially chosen ciphertexts. The concept of robustness was studied in one way or another in [18] and [17], but was only recently formalized by Abdalla *et al.* [3].

It is not hard to see that robustness can be trivially achieved by appending the encryption key to the ciphertext and checking for it upon decryption. The main drawback is that the resulting scheme is no longer anonymous. In fact, as discussed in [3] and further motivated by our results on anonymity of hybrid encryptions, it is exactly for anonymous schemes that robustness is important. In [3], the authors study the robustness properties for several existing anonymous encryption schemes, and design general constructions for transforming any IBE/PKE scheme into robust ones.

A transformation is *keyed* if an additional string needs to be added to the set of public parameters for the original scheme, and is called *non-keyed*, otherwise. An important advantage of non-keyed constructions over keyed ones is that the robustness property can be added to the encryption scheme without having to notify a third party such as a PKI in advance. Consequently, users of a system can add robustness to the scheme after it is deployed.

Non-keyed transformations for robustness. In the standard model, we design a non-keyed construction for transforming any anonymous IBE/PKE scheme into a weakly robust one in presence of CPA attacks. In the random oracle model, we design a non-keyed transformation that provides strong robustness in presence of CCA attacks. In both cases, the computational overhead is very small (it involves one to three invocations of a hash function), and despite being non-keyed the ciphertext sizes we achieve are better than those of the previous work. A curious open question is whether we can achieve the latter transformation in the standard model.

Collision-freeness. We also study the notion of *collision-freeness*, a natural relaxation of robustness. Roughly speaking, an encryption scheme is collision-free if a ciphertext does not decrypt to the same message under two different decryption keys. Collision-freeness can be a sufficient property in some scenarios in practice. For example, if the receiver expects to see a specific message as

part of the protocol but after decrypting using his secret key recovers a different one, he can detect an error and stop the communication. Interestingly, we show that schemes such as the El Gamal PKE scheme [13] and the Boyen-Waters IBE scheme [9] are strongly collision-free even though they are known not to be weakly robust. Hence, collision-freeness seems to be a less restrictive assumption on an encryption scheme and one that most encryption schemes seem to satisfy without any modifications. More importantly, we design a more efficient construction for transforming any collision-free encryption scheme to a strongly robust one.

2 Preliminaries

ONE-WAY FUNCTIONS. Roughly speaking, a function is one-way if it is hard to invert on a random input. More formally, we say that a function f over $\{0,1\}^k$ is *one-way* if

$$\mathbf{Adv}^{\mathrm{owf}}_{f,\mathsf{A}}(k) \;=\; \Pr\left[\; x \xleftarrow{\$} \{0,1\}^k \;;\; y \leftarrow f(x) \;;\; x' \xleftarrow{\$} \mathsf{A}(f,y) \;:\; x = x' \;\right]$$

is negligible for every PPT inverter A.

GENERAL ENCRYPTION SCHEMES. Abdalla *et al.* [3] introduced and used the notion of general encryption schemes which encompass both PKE and IBE schemes. Similar to their work we will use this notion, since all our transformations are applicable to both PKE and IBE schemes.

A general encryption (GE) scheme consists of a tuple $GE = (\mathsf{Pg}, \mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ of algorithms. The parameters generation algorithm Pg takes no input and returns common parameters $pars$ and a master secret key msk. On input $pars, msk, id$, the key generation algorithm Kg produces an encryption key ek and the decryption key dk. On inputs $pars, ek, M$ the encryption algorithm Enc produces a ciphertext C encrypting plaintext M. On input $pars, ek, dk, C$, the deterministic decryption algorithm Dec returns either a plaintext M or $\perp$ to indicate that it rejects. GE is a PKE scheme if $msk = \epsilon$ and Kg ignores its id input. GE is an IBE scheme if $ek = id$, meaning the encryption key generated by Kg on inputs $pars, msk, id$ is always id. Finally, we point out that the notion of general encryption contains PKE schemes, IBE schemes and more. In other words, there are general encryption schemes that are neither PKE nor IBE schemes.

AI-{CPA,CCA} SECURITY. Traditionally, the definitions of privacy [15,19,12] and anonymity [5,2] for encryption schemes are introduced separately. However, when considering robustness, it makes sense to consider both notions simultaneously. Hence we follow the definition of [3] who combine the two into a single game. We define the AI-{CPA,CCA} security (AI = ANON + IND) of a general encryption scheme $GE = (\mathsf{Pg}, \mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ via a security game between the adversary and the challenger.

- **Setup:** Challenger runs $(pars, msk) \leftarrow \mathsf{Pg}(1^k)$; $b \xleftarrow{\$} \{0,1\}$; $S, T, U, V \leftarrow \emptyset$.
- **Queries:**

- Public key query id. Challenger lets $U \leftarrow U \cup \{id\}$; $(Ek[id], Dk[id]) \xleftarrow{\$} \mathsf{Kg}(pars, msk, id)$ and returns $Ek[id]$.
- Decryption-key query id. If $id \notin U$ or $id \in S$ return $\perp$. Else $V \leftarrow V \cup \{id\}$ and return $Dk[id]$.
- Decryption query (C, id). If $id \notin U$ or $(id, C) \in T$ return $\perp$. Else let $M \leftarrow \mathsf{Dec}(pars, Ek[id], Dk[id], C)$, and return M.
- Challenge query $(id_0^*, id_1^*, M_0^*, M_1^*)$. If $id_0^* \notin U$ or $id_1^* \notin U$ or $id_0^* \in V$, or $id_1^* \in V$ return $\perp$. Else let $C^* \xleftarrow{\$} \mathsf{Enc}(pars, Ek[id_b], M_b^*)$; $S \leftarrow S \cup \{id_0^*, id_1^*\}$; $T \leftarrow T \cup \{(id_0^*, C^*), (id_1^*, C^*)\}$ and return C^*.

– **Adversary's guess.** Adversary returns a bit b'.

Note that there is only one challenge query. In case of CPA attacks, no decryption queries are allowed. Adversary A's advantage in the AI-{CPA,CCA} game is:

$$\mathbf{Adv}_{GE}^{\text{ai-}\{\text{cpa,cca}\}}(\mathsf{A}) = \Pr[b' = b] - 1/2$$

In some cases however, we consider the security notions for anonymity (ANON-{CPA,CCA}) and indistinguishability (IND-{CPA,CCA}), individually. The challenge query in the above security game can be modified in the obvious way to capture each of these definitions separately. We point out that similar definitions can also be adapted for the case of symmetric-key encryption.

Robustness. Following [3], we consider two definitions of robustness for a general encryption scheme, namely weak robustness (WROB) and strong robustness (SROB). The following game defines both notions. As noted below the only difference is in the final message sent by the adversary to the challenger:

– **Setup:** Challenger runs $(pars, msk) \leftarrow \mathsf{Pg}(1^k)$; $b \xleftarrow{\$} \{0,1\}$; $U, V \leftarrow \emptyset$.
– **Queries:**
- Public key query id. Challenger lets $U \leftarrow U \cup \{id\}$; $(Ek[id], Dk[id]) \xleftarrow{\$} \mathsf{Kg}(pars, msk, id)$ and returns $Ek[id]$.
- Decryption-key query id. If $id \notin U$ or $id \in S$ return $\perp$. Else $V \leftarrow V \cup \{id\}$ and return $Dk[id]$.
- Decryption query (C, id). If $id \notin U$ return $\perp$. Else let $M \leftarrow \mathsf{Dec}(pars, Ek[id], Dk[id], C)$, and return M.
- Final message (id_0^*, id_1^*, M) (for WROB). If $id_0 = id_1$ or $id_0^* \notin U$ or $id_1^* \notin U$ or $id_0^* \in V$, or $id_1^* \in V$ return 0. Else let $C^* \xleftarrow{\$} \mathsf{Enc}(pars, Ek[id_0], M)$; $M' \leftarrow \mathsf{Dec}(pars, Ek[id_1], Dk[id_1], C^*)$; if $M' \neq \perp$ return 1, else return 0.
- Final message (id_0^*, id_1^*, C) (for SROB). If $id_0 = id_1$ or $id_0^* \notin U$ or $id_1^* \notin U$ or $id_0^* \in V$, or $id_1^* \in V$ return 0. Else let $M_0 \leftarrow \mathsf{Dec}(pars, Ek[id_0], Dk[id_0], C)$; $M_1 \leftarrow \mathsf{Dec}(pars, Ek[id_1], Dk[id_1], C)$; if $M_0 \neq \perp$ and $M_1 \neq \perp$ return 1, else return 0.

Similar to above, in case of CPA attacks, no decryption queries are allowed. Adversary A's advantage in the {WROB,SROB}-{CPA,CCA} game is:

$$\mathbf{Adv}_{GE}^{\{\text{wrob,srob}\}\text{-}\{\text{cpa,cca}\}}(\mathsf{A}) = \Pr[G^{\mathsf{A}} \to 1]$$

In the WROB game the adversary produces a message M, and C is its encryption under the encryption key of one of the given identities, while in the SROB game adversary produces C directly, and may not obtain it as an honest encryption. Note that in case of PKE schemes, the adversary does not get to choose the encryption keys of the identities it is targeting. Those are honestly and independently chosen by the identities themselves in real life and and by the games in the above formalizations.

3 Anonymous-CCA Hybrid Encryption

In this section we take a closer look at anonymous encryption schemes in presence of *chosen ciphertext attacks* (ANON-CCA) as defined in Section 2. Previous works on anonymous public-key and identity-based encryption [5,2] have studied this security notion and provided constructions satisfying it.

However, in most scenarios in practice, PKE and IBE schemes are used in the KEM/DEM paradigm. It is known that if the KEM component is IND-CCA secure and the DEM component is (one-time) IND-CCA, the resulting hybrid encryption is also IND-CCA secure. For practical reasons, it is crucial to determine whether we can make similar statements when considering the anonymity of the resulting hybrid construction. More specifically, we try to answer the following question:

Given an ANON-CCA PKE or IBE scheme and an (ANON-CCA + IND-CCA) symmetric-key encryption scheme, is the resulting hybrid encryption scheme ANON-CCA?

3.1 A Negative Result

Somewhat surprisingly, we answer the above question in the negative. First we show a counterexample by building an ANON-CCA secure PKE/IBE scheme and a symmetric-key IND-CCA encryption, where it is easy to break the anonymity of the resulting hybrid construction. The negative result easily extends to the case when the symmetric-key component is also ANON-CCA. An important implication is that *designing ANON-CCA PKE or IBE schemes is not sufficient for providing anonymity in practice where, more often than not, encryption schemes are used in hybrid constructions.*

Claim 31 *There exist an ANON-CCA PKE/IBE scheme and a symmetric-key authenticated encryption scheme (assuming there are secure schemes at all) such that the resulting hybrid encryption is not ANON-CCA.*

The intuition behind the counterexample is that since the adversary has access to a decryption oracle, he can take advantage of the fact that decrypting one ciphertext under two different secret keys can result in different answers. Particularly, these different answers can be used by the adversary to compromise the anonymity of the scheme.

Proof. We describe the proof for the case of a PKE scheme, but an identical proof works for IBE schemes as well. Let $\mathsf{PKE}_1 = (\mathsf{Kg}_1, \mathsf{Enc}_1, \mathsf{Dec}_1)$ be an (ANON-CCA + WROB-CCA) PKE encryption scheme. The Cramer-Shoup encryption scheme or any of the constructions in this paper will do. We build the encryption scheme $\mathsf{PKE}_2 = (\mathsf{Kg}_2, \mathsf{Enc}_2, \mathsf{Dec}_2)$ by letting the key-generation and encryption algorithms be identical to those of PKE_1, and modifying the decryption algorithm such that whenever the Dec_1 algorithm returns the symbol $\perp$, the decryption algorithm Dec_2 returns 0^n instead, and otherwise works similar to Dec_1. It is easy to verify that after this simple modification, PKE_2 remains ANON-CCA. PKE_2 will be the key encapsulation method in our counterexample.

For the DEM component we use an IND-CCA encryption scheme that is also *key-binding*, a notion introduced in [14].

Definition 1. *A symmetric-key encryption scheme $\mathcal{E} = (\mathcal{SK}, \mathcal{SE}, \mathcal{SD})$ is called key-binding if for any key k generated by $\mathcal{SK}$, any message m, and randomness r, there does not exist a key k' such that $k' \neq k$ and $\mathcal{SD}_{k'}(\mathcal{SE}_k(m, r)) \neq \perp$.*

The key-binding property guarantees that a ciphertext created using one secret key, does not decrypt correctly under any other secret key. Fischlin [14] showed simple constructions of such encryption schemes from any PRF. For the purpose of our counterexample it suffices to know that an IND-CCA encryption scheme $\mathcal{E}$ with such a property exists.

Now, we show that combining PKE_2 and $\mathcal{E}$ into a hybrid encryption is not ANON-CCA. Particularly, an attacker with the following simple strategy can break the anonymity of the scheme.

Recall the ANON-CCA security game. Attacker A initially sends a message m as his challenge in the ANON-CCA game and receives the ciphertext $C = (c_1, c_2) = (\mathsf{Enc}(pk_{id_b}, k), \mathcal{SE}_k(m))$ for a random bit $b \in \{0, 1\}$ and a random key $k \in \{0, 1\}^n$. Then, A makes a decryption query for the ciphertext $(c_1, \mathcal{SE}_{0^n}(m'))$ under public key pk_{id_0}, for an arbitrary message m'. If the answer is $\perp$, A outputs 0 and else outputs 1.

To see why A breaks the ANON-CCA security of the encryption scheme note that if $b = 1$ then $k' = \mathsf{Dec}_2(sk_{id_1}, \mathsf{Enc}_2(pk_{id_0}, k)) = 0^n$ given the way we have defined PKE_2. Hence, we have that $\mathcal{SD}_{0^n}(\mathcal{SE}_{0^n}(m')) = m' \neq \perp$. On the other hand if $b = 0$ then $k' = \mathsf{Dec}_2(sk_{id_0}, \mathsf{Enc}_2(pk_{id_0}, k)) = k$. Hence we have $\mathcal{SD}_k(\mathcal{SE}_{0^n}(m')) = \perp$ due to the key-binding property of $\mathcal{E}$ and the fact that $k \neq 0^n$ with all but negligible probability. Therefore, A guesses the bit b correctly with high probability.

A closer look at the above attack strategy reveals that a much weaker property than that of definition 1 for the symmetric-key scheme suffices for our argument to go through. In particular, we only need the *key binding* property to hold for a fixed message and a fixed secret key (m' and 0^n, respectively).

Strengthening the DEM component? One potential solution is to use a symmetric-key encryption scheme that possesses some additional properties.

Particularly, one natural question is whether using an anonymous-CCA symmetric-key encryption as the DEM component would yield an anonymous hybrid construction. Unfortunately, the answer to this question is also negative. It is easy to verify that the above negative result extends to work for any security notion considered for symmetric-key encryption, as long as that security notion can be achieved in conjunction with the *key-binding* property. In all such cases, the proof given above works without any significant changes.

Anonymity of symmetric-key encryption schemes has been studied under the name *key-hiding* in [14] where the authors also design IND-CCA secure symmetric-key encryption schemes that are simultaneously key-hiding and key-binding. This leads to the following claim:

Claim 32 *There exist an ANON-CCA PKE/IBE scheme and an (ANON-CCA + IND-CCA) symmetric-key encryption scheme such that the resulting hybrid encryption is not ANON-CCA.*

3.2 A Positive Result

In light of the above negative results, it is natural to ask what additional property the KEM component should have in order to preserve its ANON-CCA security in a hybrid construction. We show that if one further assumes that the KEM component is *weakly-robust*, the resulting hybrid encryption is in fact ANON-CCA. This implies that despite the negative results we gave above, for most ANON-CCA schemes we know such as the Boneh-Franklin IBE, the Cramer-Shoup PKE, and the DHIES PKE all of which are known to be weakly-robust [3], using them as part of a hybrid construction is safe. The intuition behind the proof is that weak robustness ensures that the decryption algorithm behaves in a *predictable* way, when decrypting a ciphertext under two different secret keys, and this predictable behavior combines quite nicely with the security properties of an authenticated symmetric encryption scheme, namely, IND-CCA security and the ciphertext integrity (CTXT-INT).

In the following claim we prove a stronger result than what we need here by considering the notion of AI-CCA security which combines ANON-CCA security and IND-CCA security into one definition. The main reason is that we need this stronger claim in a following section. The proof for the case when one is only interested in ANON-CCA secure hybrid schemes is identical.

Claim 33 *If the KEM component* PKE *of a hybrid construction is an (AI-CCA + WROB-CCA) general encryption, and* $\mathcal{E}$ *is a one-time authenticated symmetric encryption, then the resulting hybrid encryption* PKE' *is also an AI-CCA general encryption scheme.*

Proof. We prove the above claim via a sequence of games.

Game 0. Game 0 is simply the AI-CCA game. Denote by b the random bit generated by the challenger, by C^* the challenge ciphertext $C^* = (c_1^*, c_2^*)$ where c_1^* is the KEM component and c_2^* is the DEM component, and by k^* the secret key used for the DEM component.

Game 1. Game 1 is similar to game 0, except that for any decryption queries of the form (c_1, c_2) for pk_{id_b} where $c_1 = c_1^*$ and $c_2 \neq c_2^*$, challenger uses k^* to decrypt c_2 and recover the message (as opposed to decrypting c_1).

It is easy to see that the difference between the advantage of any adversary in these two games is bounded by the decryption error. For simplicity we assume that there is no decryption error and therefore

$$\mathbf{Adv}_{G_0}(\mathsf{A}) = \mathbf{Adv}_{G_1}(\mathsf{A})$$

Game 2. Similar to game 1 except that for any decryption queries of the form (c_1, c_2) for $pk_{id_{1-b}}$ where $c_1 = c_1^*$ and $c_2 \neq c_2^*$), challenger returns $\perp$.

Note that games 1 and 2 are different only when c_1^* which is the encryption of message m_b under pk_{id_b}, also decrypts correctly under the $pk_{id_{1-b}}$. This probability is bounded by the advantage of an adversary B in winning the WROB-CCA game and hence:

$$\mathbf{Adv}_{G_2}(\mathsf{A}) - \mathbf{Adv}_{G_1}(\mathsf{A}) \leq \mathbf{Adv}^{\mathrm{wrob-cca}}_{\mathsf{PKE}}(\mathsf{B})$$

Game 3. Similar to game 2 except that the challenger generates and uses a random key k' (instead of k^*) when encrypting the private-key component of the ciphertext for the challenge query.

The difference between the advantages of an adversary in games 2 and 3 is bounded by the AI-CCA security of the PKE scheme:

$$\mathbf{Adv}_{G_3}(\mathsf{A}) - \mathbf{Adv}_{G_2}(\mathsf{A}) \leq \mathbf{Adv}^{\mathrm{ai-cca}}_{\mathsf{PKE}}(\mathsf{B}')$$

Game 4. We modify game 3 in two ways. First, for the challenge query, instead of encrypting the message m_b, the challenger encrypts the constant message 0^k. Second, for decryption queries (c_1, c_2) under pk_{id_b} where $c_1 = c_1*$ the challenger returns $\perp$.

The probability of distinguishing the first change is bounded by the IND-CCA advantage of an adversary against the $\mathcal{E}$ scheme, while for second change, the probability is bounded by the advantage of an adversary playing the ciphertext integrity (CTXT-INT) game with $\mathcal{E}$. Both the IND-CCA security and the CTXT-INT security are properties that are possessed by any authenticated encryption scheme.

$$\mathbf{Adv}_{G_4}(\mathsf{A}) - \mathbf{Adv}_{G_3}(\mathsf{A}) \leq \mathbf{Adv}^{\mathrm{ind-cca}}_{\mathcal{E}}(\mathsf{B}'') + \mathbf{Adv}^{\mathrm{ctxt-int}}_{\mathcal{E}}(\mathsf{B}''')$$

Finally, it is easy to see that the adversary's view in game 4 is independent of the bit b and hence adversary's advantage in guessing b is exactly $1/2$. Putting things together we have:

$$\mathbf{Adv}^{\mathrm{ai-cca}}_{\mathsf{PKE}'}(\mathsf{A}) \leq \mathbf{Adv}^{\mathrm{wrob-cca}}_{\mathsf{PKE}}(\mathsf{B}) + \mathbf{Adv}^{\mathrm{ai-cca}}_{\mathsf{PKE}}(\mathsf{B}') + \mathbf{Adv}^{\mathrm{ind-cca}}_{\mathcal{E}}(\mathsf{B}'') + \mathbf{Adv}^{\mathrm{ctxt-int}}_{\mathcal{E}}(\mathsf{B}''')$$

4 Non-keyed Transformations for Robustness

Having further motivated the study of robust encryption schemes, we next focus on efficient ways of transforming general encryption schemes into robust ones. As mentioned earlier, such a transformation is called a keyed transformation if an additional string is added to the original set of public parameters, and is called non-keyed otherwise.

In Section 4.1, we design an efficient and non-keyed transformation for weak-robustness, in presence of CPA attacks (in the standard model). In Section 4.2, we design a non-keyed transformation for strong-robustness in presence of CCA attacks (in the random oracle model). Despite being non-keyed, our transformations have better ciphertext sizes compared to previous work. In other words, not adding an extra string to the public parameters does translate to larger ciphertexts (see the efficiency comparison sections).

4.1 A Transformation for AI-CPA Schemes

The following non-keyed construction takes any AI-CPA encryption scheme, and transforms it to a (AI-CPA + WROB-CPA) scheme.

Construction 41 Let $\mathsf{PKE} = (\mathsf{Pg}, \mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a AI-CPA general encryption scheme, and let f be a one-way function over $\{0,1\}^k$. We construct the general encryption scheme $\mathsf{PKE}' = (\mathsf{Pg}', \mathsf{Kg}', \mathsf{Enc}', \mathsf{Dec}')$:

- **Parameter Generation**(Pg'): On input 1^k return $(pars, msk) \xleftarrow{\$} \mathsf{Pg}(1^k)$.
- **Key Generation**(Kg'): On input $pars, msk, id$, return $(pk_{id}, sk_{id}) \xleftarrow{\$} \mathsf{Kg}(pars, msk, id)$.
- **Encryption**(Enc'): On input $pars, pk_{id}, m$, generate a random $r \in \{0,1\}^k$ and return $(\mathsf{Enc}(pars, pk_{id}, m||r), f(r))$.
- **Decryption**(Dec'): On inputs $pars, pk_{id}, sk_{id}, (c_1, c_2)$, compute $m'||r' \xleftarrow{\$} \mathsf{Dec}(pars, pk_{id}, sk_{id}, c_1)$. If $r' \neq \perp$ and $f(r') = c_2$ return m'; else return $\perp$.

Note that in construction 41, instead of a one-way function, we can also use a target collision-resistant (TCR) hash function (a universal one-way hash function). Particularly, it is easy to show that any TCR function that is sufficiently compressing is a good one-way function.

We will shortly prove the security of the above scheme, but first lets briefly study its efficiency. Efficiency comparison. To implement our scheme one can use a fixed-length cryptographic hash function h with output length of 128 bits (e.g. constructed by suitably modifying the output length of a hash function from the SHA family). The reason that we only need 128 bits of output is that we only require the hash function to be one-way as opposed to collision-resistant. Furthermore, it is sufficient for us to let $k = 256$ where r is chosen from $\{0,1\}^k$.[2] This means that the PKE scheme has to encrypt a message that is only 256 bits

[2] When computing hash of r, we can pad r with enough 0's in order to match the input block-size requirement for the hash function. Note that this does not effect the efficiency of the encryption or the size of ciphertext in any way.

longer than the original message and the ciphertext is at most expanded by an additive factor of 384 bits as opposed to 768 bits in construction of Abdalla *et al.* [3].

Theorem 1. *Let* PKE *be a AI-CPA secure general encryption scheme and* f *be a one-way function. Then, the* PKE' *scheme of construction 41 is both AI-CPA secure and WROB-CPA secure.*

Proof. We prove the theorem in two separate claims. Claim 42 ensures that the above transformation preserves the AI-CPA security of the original scheme. Claim 43 states that the resulting scheme PKE' is also weakly robust.

Claim 42 *For any PPT adversary* A *against* PKE'*, there exist a PPT adversary* B *against* PKE *such that:*

$$\mathbf{Adv}^{\text{ai-cpa}}_{\mathsf{PKE}'}(A) = \mathbf{Adv}^{\text{ai-cpa}}_{\mathsf{PKE}}(B)$$

Proof. B runs A. When A sends its challenge request (id_0, id_1, M_0, M_1), B generates a random value $r \in \{0,1\}^k$ and sends $(id_0, id_1, M_0||r, M_1||r)$ to its own challenger in the AI-CPA game for PKE. B receives back $c^* = \mathsf{Enc}(pars, id_b, pk_{id_b}, M_b||r)$ and sends $(c^*, f(r))$ to A. The decryption-key queries made by A are forwarded to the corresponding oracle in B's game. Since we only consider CPA attacks, no decryption queries on id_0 or id_1 are allowed. Eventually, A outputs a bit b'. B also outputs b' and halts. It is straightforward to see that the advantage of B against the PKE is the same as A's advantage against the PKE' scheme.

Claim 43 *For any PPT adversary* A *against the* PKE' *in the WROB-CPA game, there exist PPT adversaries* B_1 *against the* PKE *in the AI-CPA game and* B_2 *against* f *in the one-wayness game such that:*

$$\mathbf{Adv}^{\text{wrob-cpa}}_{\mathsf{PKE}'}(A) = 2\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{PKE}}(\mathsf{B}_1) + \mathbf{Adv}^{\text{owf}}_{f}(\mathsf{B}_2)$$

Proof. We prove this claim in a sequence of two games.

Game 0. Game 0 is the WROB-CPA game against the PKE' scheme as defined earlier. More specifically, adversary sends the tuple (id_0, id_1, M) to the challenger. Challenger computes $C_0 = \mathsf{Enc}'(pars, id_0, pk_{id_0}, M) = (\mathsf{Enc}(pars, pk_{id_0}, M||r), f(r))$ for random $r \in \{0,1\}^k$. He then computes $M_1 = \mathsf{Dec}'(pars, pk_{id_1}, sk_{id_1}, C_0)$. If $M_1 \neq \perp$, the challenger outputs 1. Else it outputs 0.

Game 1. Game 1 is similar to game 0, except that C_0 is computed in the following way:

$$C_0 = (\mathsf{Enc}(pars, pk_{id_0}, M||0^k), f(r))$$

The rest of the game stays the same.

First we show that there exist an adversary B_1 such that $\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{PKE}}(\mathsf{B}_1) = 1/2(\Pr[G^A_1 \to 1] - \Pr[G^A_0 \to 1])$. B_1 runs A and receives the tuple (id_0, id_1, M) from her. B_1 queries the key oracle for (pk_{id_1}, sk_{id_1}). He then generates a random

$r \in \{0,1\}^k$ and sends $(id_0, m'_0 = M||r, m'_1 = M||0^k)$ to the challenger in the IND-CPA game against PKE and receives $C_0 = \mathsf{Enc}(pars, id_0, pk_{id_0}, m'_b)$ for a random bit b. B_1 then decrypts $(C_0, f(r))$ using the Dec' algorithm and the secret key sk_{id_1}. If the result of decryption is not $\perp$, B_1 lets $b' = 1$ and else $b' = 0$. Then we have

$$\begin{aligned}
&\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{PKE}}(\mathsf{B}_1) = \Pr[b' = b] - 1/2 = \\
&\Pr[b = 1] \cdot \Pr[\mathsf{B}_1^{\text{ind-cpa}} \to 1 | b = 1] + \Pr[b = 0] \cdot \Pr[\mathsf{B}_1^{\text{ind-cpa}} \to 0 | b = 0] = \\
&1/2 \Pr[\mathsf{B}_1^{\text{ind-cpa}} \to 1 | b = 1] + 1/2 \Pr[\mathsf{B}_1^{\text{ind-cpa}} \to 0 | b = 0] - 1/2 = \\
&1/2 \Pr[G_1^{\mathsf{A}} \to 1] + 1/2(1 - \Pr[G_0^{\mathsf{A}} \to 1]) - 1/2 = \\
&1/2(\Pr[G_1^{\mathsf{A}} \to 1] - \Pr[G_0^{\mathsf{A}} \to 1])
\end{aligned} \tag{1}$$

We now show that there exist an adversary B_2 such that $\Pr[G_1^{\mathsf{A}} \to 1] = \mathbf{Adv}^{\text{owf}}_f(\mathsf{B}_2)$. B_2 generates the $(pars, msk)$ for the general encryption, and runs A. When B_2 receives the tuple (id_0, id_1, M) he computes (pk_{id_0}, sk_{id_0}), (pk_{id_1}, sk_{id_1}) and $C_0 = \mathsf{Enc}(pars, pk_{id_0}, M_0||0^k)$. He then requests his challenge for the one-wayness game and receives $f(r)$ for a random r. B_2 then decrypts using $(C_0, f(r))$ using the Dec' algorithm and the secret key sk_{id_1}. If the result is $\perp$ it outputs fail and halts. Else, it parses the decrypted plaintext into $M'||r'$ and returns r' to his own challenger.

B_2 wins the one-wayness game if $f(r') = f(r)$. Note that according to the definition of Dec', whenever the decryption algorithm does not output $\perp$ we have $f(r') = f(r)$. Hence

$$\mathbf{Adv}^{\text{owf}}_f(\mathsf{B}_2) = 1 - \Pr[\mathsf{B}_{2f}^{\text{owf}} \to \mathsf{fail}] = 1 - \Pr[{G_1}^{\mathsf{A}} \to 0] = \Pr[{G_1}^{\mathsf{A}} \to 1]$$

Putting things together we have:

$$\begin{aligned}
&\mathbf{Adv}^{\text{wrob-cpa}}_{\mathsf{PKE}'}(\mathsf{A}) = \Pr[G_0^{\mathsf{A}} \to 1] = \\
&\Pr[G_0^{\mathsf{A}} \to 1] - \Pr[G_1^{\mathsf{A}} \to 1] + \Pr[G_1^{\mathsf{A}} \to 1] = \\
&2\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{PKE}}(\mathsf{B}_1) + \mathbf{Adv}^{\text{owf}}_f(\mathsf{B}_2)
\end{aligned}$$

4.2 A Transformation for AI-CCA Schemes

Unfortunately, the transformation we gave above does not work in case of AI-CCA encryption schemes. Nevertheless, we are able to design an efficient and non-keyed transformation for any AI-CCA encryption scheme, in the random oracle model. The construction follows:

Construction 44 Let $\mathsf{PKE} = (\mathsf{Pg}, \mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be an AI-CCA general encryption scheme, and let $G, H, H' : \{0,1\}^* \to \{0,1\}^k$ be three hash functions. We construct the general encryption scheme $\mathsf{PKE}' = (\mathsf{Pg}', \mathsf{Kg}', \mathsf{Enc}', \mathsf{Dec}')$:

- **Parameter Generation**(Pg'): On input 1^k return $(pars, msk) \xleftarrow{\$} \mathsf{Pg}(1^k)$.
- **Key Generation**(Kg'): On input $pars, msk, id$, return $(pk_{id}, sk_{id}) \xleftarrow{\$} \mathsf{Kg}(pars, msk, id)$.
- **Encryption**(Enc'):
  On input $pars, pk_{id}, m$, generate a random $r \in \{0,1\}^k$ and return $(\mathsf{Enc}(pars, pk_{id}, r; H(r)), G(r) \oplus m, H'(pk, r, m))$.
- **Decryption**(Dec'): On inputs $pars, pk_{id}, sk_{id}, (c_1, c_2, c_3)$, compute $r' \xleftarrow{\$} \mathsf{Dec}(pars, pk_{id}, sk_{id}, c_1)$. If $r' = \perp$ or $\mathsf{Enc}(pars, pk_{id}, r'; H(r')) \neq c_1$, return $\perp$, else compute $m \leftarrow c_2 \oplus G(r)$; if $H'(pk, r, m) = c_3$ return m, else return $\perp$.

The above construction is an adaptation of an earlier version of the OAEP scheme (see [6]) based on any one-way trapdoor function (TDF). The two main differences are that (i) we are transforming a randomized encryption scheme instead of a one-way TDF which is why we use $H(r)$ to generate the randomness for the encryption algorithm, and (ii) since our goal is to also achieve robustness, the third component of the ciphertext hashes the public key along with the message and randomness.

It is also interesting to note that unlike the optimized OAEP scheme [7] which encrypts $c_2||c_3$ as part of the message (in order to obtain shorter ciphertexts), due to the impossibility result of [3] who rule out non-keyed redundancy codes, there is no hope of doing the same in our case.

EFFICIENCY COMPARISON. The overhead for the ciphertext size is two hash values each of which leads to 512 bits of overhead. The alternative existing solution would be to combine a weakly robust encryption scheme with the weak-to-strong transformation of [3]. This leads to $768 + x$ bits where the x is the ciphertext overhead of the weak-to-strong transformation which can be quite large itself (depending on the commitment scheme used).

Theorem 2. *Let* PKE *be an AI-CCA secure general encryption scheme and* H, G*, and* H' *be random oracles. Then, the* PKE' *scheme of construction 44 is both AI-CCA secure and SROB-CCA secure.*

We prove the above theorem via two separate claims. Claim 45 ensures that the above transformation preserves the AI-CCA security of the original scheme. Claim 46 states that the resulting scheme PKE' is also weakly robust.

Claim 45 *For any PPT adversary* A *against* PKE'*, there exist a PPT adversary* B *against* PKE *such that:*

$$\mathbf{Adv}^{\text{ai-cca}}_{\mathsf{PKE}'}(\mathsf{A}) < q_D/2^k + q_H \mathbf{Adv}^{\text{ai-cca}}_{\mathsf{PKE}}(\mathsf{B})$$

Proof. We prove this claim in a sequence of games.

Game 0. In this game the adversary plays the AI-CCA game with the challenger using the construction above. The challenger initializes three empty lists H_{list}, G_{list}, and H'_{list}. For any oracle query q made to H (G, or H'), if a tuple of the form (q, a) for any a is present in H_{list} (G_{list} or H'_{list}) returns a as the answer. Else, challenger generates a random $a \in \{0,1\}^k$, adds (q, a)

to the H_{list} (G_{list} or H'_{list}) and returns a to the adversary. Denote the adversary's challenge query by (m_0, m_1, id_0, id_1), and the response ciphertext by $c^* = (c_1^*, c_2^*, c_3^*) = (\mathsf{Enc}(pars, pk_{id_b}, r; H(r)), G(r) \oplus m_b, H'(pk_{id_b}, r, m_b))$ for a random bit $b \in \{0,1\}$ and $r \in \{0,1\}^k$. Decryption queries are answered by the challenger using the decryption algorithm described above. Adversary eventually outputs the bit b' and wins if $b' = b$. For any PPT adversary A we have

$$\mathbf{Adv}^{\text{ai-cca}}_{\mathsf{PKE}'}(\mathsf{A}) = \mathbf{Adv}_{G_0}(\mathsf{A}) - 1/2$$

Game 1. Similar to game 0, except that on decryption queries of the form $c = (c_1, c_2, c_3)$ where $c_1 = c_1^*$, if there exist a tuple of the form $(q, c_3) \in H'_{list}$, challenger parses $(pk, r, m) \leftarrow q$, and recomputes the first two components of the ciphertext using these values. If they match c_1 and c_2 sent by the adversary, it returns m. If the values do not match or the tuple of the form (q, c_3) does not exist, challenger returns $\perp$.

A's view in the two games is different only in the case that he has not queried q to the list but is able to guess $c_3 = H'(q)$. This only happens with probability $1/2^k$ for every decryption query. Hence

$$\mathbf{Adv}_{G_0}(\mathsf{A}) - \mathbf{Adv}_{G_1}(\mathsf{A}) \leq q_D/2^k$$

Game 2. This game is identical to game 1 except that if A makes an oracle query for H or G on input r where r is the random message encrypted in the challenge ciphertext, the challenger outputs $fail$ and ends the game.

Based on the fundamental lemma game playing we have

$$\mathbf{Adv}_{G_1}(\mathsf{A}) - \mathbf{Adv}_{G_2}(\mathsf{A}) \leq \Pr[G_1^{\mathsf{A}} \rightarrow fail]$$

Next we will bound the probability of outputting fail, by the advantage of an adversary B who the one-way-CCA game against the PKE scheme. We show that for any adversary A winning the game G_2, there exist a PPT adversary B winning the one-way-CCA game against the original scheme PKE.

B generates a random index $i \in [1..q_H]$. B then runs A. When A makes his challenge query (m_0, m_1, id_0, id_1), B generates a random bit b, and asks for his challenge ciphertext under id_b to receive $c_1^* = \mathsf{Enc}(pk_{id_b}, r)$ for a random message r. B computes c_2^* and c_3^* on his own and replies to A with (c_1^*, c_2^*, c_3^*).

On an oracle query a (for any of the three oracles), if this is the ith oracle query, B outputs a to his own challenger and halts. Else, if a was queried before, he returns the same answer, and if not, he generates a random answer and adds the tuple to the corresponding list.

On a decryption query (c_1, c_2, c_3) where $c_1 \neq c_1^*$, B uses his own decryption oracle for Dec and performs the Dec' decryption algorithm. Here, it is critical for the randomness used in the encryption algorithm to be derivable from the decrypted message, and this is why $H(\cdot)$ is used as the randomness (or else B would not be able to perform the verification component of Dec'). For any decryption query (c_1, c_2, c_3) where $c_1 = c_1^*$, B performs exactly what the challenger in game 1 does. It is easy to see that

$$\Pr[G_2^{\mathsf{A}} \rightarrow fail] \leq q_H \mathbf{Adv}^{\text{ow-cca}}_{\mathsf{PKE}}(\mathsf{B}) \leq q_H \mathbf{Adv}^{\text{ai-cca}}_{\mathsf{PKE}}(\mathsf{B})$$

.

For any adversary A who makes an oracle query for the challenge random message, there is an adversary B' who does not make such a query and has a better advantage (since such a query does not help the adversary win)

$$\mathbf{Adv}_{G_2}(\mathsf{A}) \leq \mathbf{Adv}_{G_2}(\mathsf{B}')$$

Finally, given that B' does not query r to the oracle, the challenge ciphertext is completely independent of the challenge bit b and hence

$$\mathbf{Adv}_{G_2}(\mathsf{B}') = 1/2$$

Putting everything together we have:

$$\mathbf{Adv}^{\text{ai-cca}}_{\mathsf{PKE}'}(\mathsf{A}) < q_D/2^k + q_H \mathbf{Adv}^{\text{ai-cca}}_{\mathsf{PKE}}(\mathsf{B})$$

Claim 46 *For any adversary* A *against* PKE' *we have* $\mathbf{Adv}^{\text{srob-cca}}_{\mathsf{PKE}'}(\mathsf{A}) \leq 1/2^k$.

Proof. The proof of the above claim is simple. The main observation is that a ciphertext c_1, c_2, c_3 is valid under two different public keys only if $H'(pk, \cdot, \cdot) = H'(pk', \cdot, \cdot)$ where $pk \neq pk'$. But this only happens with probability $1/2^k$ due to the fact that H' is a random oracle.

5 Collision-free Encryption and Robustness

In this section we introduce the notion of *collision-freeness*, a natural relaxation of the notion of robustness for general encryption schemes. Intuitively, collision-freeness requires that a ciphertext decrypts to two *different* plaintexts when decrypted using distinct secret keys. Our main motivation is to use collision-freeness as a stepping stone for designing robust encryption schemes. Particularly, we design a more efficient construction for transforming collision-free encryption schemes to strongly robust ones. However, we also believe that collision-freeness is a sufficient property in some scenarios in practice.

Similar to the notion of robustness, we consider weak and strong collision-freeness (WCFR and SCFR). Interestingly, we show that schemes such as the El Gamal PKE scheme [13] and the Boyen-Waters IBE scheme [9] are strongly collision-free even though they are known not to be even weakly robust. Hence, collision-freeness seems to be a less restrictive assumption on an encryption scheme and one that most encryption schemes seem to satisfy without any modifications. The following security game defines the two variants:

- **Setup:** Challenger runs $(pars, msk) \leftarrow \mathsf{Pg}(1^k)$; $b \xleftarrow{\$} \{0,1\}$; $U, V \leftarrow \emptyset$.
- **Queries:**
  - Public key query id. Challenger lets $U \leftarrow U \cup \{id\}$; $(Ek[id], Dk[id]) \xleftarrow{\$} \mathsf{Kg}(pars, msk, id)$ and returns $Ek[id]$.

- Decryption-key query id. If $id \notin U$ or $id \in S$ return $\perp$. Else $V \leftarrow V \cup \{id\}$ and return $Dk[id]$.
- Decryption query (C, id). If $id \notin U$ return $\perp$. Else let $M \leftarrow \mathsf{Dec}(pars, Ek[id], Dk[id], C)$, and return M.
- Final message (id_0^*, id_1^*, M) (for WCFR). If $id_0 = id_1$ or $id_0^* \notin U$ or $id_1^* \notin U$ or $id_0^* \in V$, or $id_1^* \in V$ return 0. Else let $C^* \xleftarrow{\$} \mathsf{Enc}(pars, Ek[id_0], M)$; $M' \leftarrow \mathsf{Dec}(pars, Ek[id_1], Dk[id_1], C^*)$; if $M' = M$ return 1, else return 0.
- Final message (id_0^*, id_1^*, C) (for SCFR). If $id_0 = id_1$ or $id_0^* \notin U$ or $id_1^* \notin U$ or $id_0^* \in V$, or $id_1^* \in V$ return 0. Else let $M_0 \leftarrow \mathsf{Dec}(pars, Ek[id_0], Dk[id_0], C)$; $M_1 \leftarrow \mathsf{Dec}(pars, Ek[id_1], Dk[id_1], C)$; if $M_0 = M_1$ return 1, else return 0.

In case of CPA attacks, no decryption queries are allowed. Adversary A's advantage in the {WCFR,SCFR}-{CPA,CCA} game is:

$$\mathbf{Adv}_{GE}^{\{\text{wcfr,scfr}\}-\{cpa,cca\}}(\mathsf{A}) = \Pr[G^{\mathsf{A}} \rightarrow 1]$$

Collision-freeness of an encryption scheme can be a sufficient requirement in some scenarios in practice. For example, if the receiver expects to see a specific message as part of the protocol but after decrypting using his secret key recovers a different one, he can detect an error and stop the communication. This makes collision-freeness a particularly attractive definition, since most of the existing anonymous encryption schemes, already satisfy this property without any additional modifications. The following claim mentions two well-known encryption schemes both of which are known not to be weakly-robust but which are collision-free.

Claim 51 *The El Gamal PKE scheme and the Boyen-Waters anonymous IBE scheme are SCFR-CPA scheme.*

The proof of the above claim quite simple but is omitted due to lack of space. Next we give a construction for transforming any strongly collision-free AI-CPA scheme into a strongly robust one. First we use the collision-free encryption scheme PKE to encrypt a random message r. Then, we hash the random message using a compressing collision resistant hash function h. We then use a strong extractor (e.g. a universal hash function) to extract the remaining randomness in r and use it as the key to a one-time symmetric-key encryption scheme.

The intuition is that (1) the collision-freeness of the PKE and the collision-resistance of the hash function h combined imply the strong robustness of the resulting scheme. More specifically, it is not hard to show that given any adversary that breaks the strong robustness of PKE', there exist an adversary that finds a collision for h: The collision-finding adversary decrypts the same ciphertext using the secret keys for two different public keys (identities) and outputs the two plaintexts as his collision for the hash function. The collision-freeness of the PKE ensures that the two plaintexts are different with high probability. (2) Given that r is chosen uniformly at random, PKE is IND-CPA secure, and

$h(r)$ only leaks a fraction of bits of r, we can use the leftover hash lemma [16] to extract most of the remaining randomness and use it as the secret key to the symmetric-key encryption scheme.

Construction 52 Let $\mathsf{PKE} = (\mathsf{Pg}, \mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be a (SCFR-CPA + AI-CPA) general encryption scheme; $h : \{0,1\}^{\ell_1} \to \{0,1\}^{\ell_2}$ be a collision-resistant hash function; $Ext : \{0,1\}^k \times \{0,1\}^{\ell_1} \to \{0,1\}^{\ell_3}$ be a family of pairwise independent hash functions, where $\ell_3 \approx \ell_1 - \ell_2$; and $\mathcal{E} = (\mathcal{SK}, \mathcal{SE}, \mathcal{SD})$ be a one-time IND-CPA symmetric-key encryption scheme. We construct the general encryption scheme $\mathsf{PKE}' = (\mathsf{Pg}', \mathsf{Kg}', \mathsf{Enc}', \mathsf{Dec}')$:

- **Parameter Generation**: On input 1^k return $(pars, msk) \xleftarrow{\$} \mathsf{Pg}(1^k)$.
- **Key Generation**: On input $pars, msk, id$, return $(pk_{id}, sk_{id}) \xleftarrow{\$} \mathsf{Kg}(pars, msk, id)$.
- **Encryption**: On input $pars, pk_{id}, m$, generate a random $r \in \{0,1\}^{\ell_1}$ and $K \in \{0,1\}^k$ and return $(\mathsf{Enc}(pars, pk_{id}, r), h(r),\ K, \mathcal{SE}(Ext(K, r), m))$.
- **Decryption**: On inputs $pars, pk_{id}, sk_{id}, (c_1, c_2, c_3)$, compute $r' \xleftarrow{\$} \mathsf{Dec}(pars, pk_{id}, sk_{id}, c_1)$. If $h(r') = c_2$ return $m' \leftarrow \mathcal{SD}(Ext(K, r'), c_3)$, else return $\perp$.

The following theorem summarizes the result. Due to lack of space, we defer the proof to the full version of the paper.

Theorem 3. *Let* PKE *be a (AI-CPA + SCFR-CPA) secure general encryption scheme, h be a CRHF, Ext be a pairwise independent hash function and $\mathcal{E}$ be a one-time IND-CPA symmetric-key encryption scheme. Then, the* PKE' *scheme of construction 52 is both AI-CPA secure and SROB-CPA secure.*

Efficiency and Comparison. The computational overhead for the transformation is negligible as it includes one invocation of a collision-resistant hash function and a pairwise-independent hash function. As an alternative to the above construction, one could also combine the construction 41, which leads to a weakly robust encryption, with the weak-to-strong-robustness transformations of [3] to achieve the same goal. However, the resulting transformations are less efficient than the above transformation since we also took advantage of the collision-freeness of the encryption scheme. Furthermore, since all the encryption schemes we know of seem to possess the collision-freeness property, the improved efficiency comes for "free".

References

1. Martín Abadi and Phillip Rogaway. Reconciling two views of cryptography (the computational soundness of formal encryption). 20(3):395, July 2007.
2. Michel Abdalla, Mihir Bellare, Dario Catalano, Eike Kiltz, Tadayoshi Kohno, Tanja Lange, John Malone-Lee, Gregory Neven, Pascal Paillier, and Haixia Shi. Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions. 21(3):350–391, July 2008.

3. Michel Abdalla, Mihir Bellare, and Gregory Neven. Robust encryption. In *TCC*, pages 480–497, 2010.
4. Michel Abdalla, Mihir Bellare, and Phillip Rogaway. The oracle Diffie-Hellman assumptions and an analysis of DHIES. pages 143–158, 2001.
5. Mihir Bellare, Alexandra Boldyreva, Anand Desai, and David Pointcheval. Key-privacy in public-key encryption. pages 566–582, 2001.
6. Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. pages 62–73, 1993.
7. Mihir Bellare and Phillip Rogaway. Optimal asymmetric encryption. pages 92–111, 1994.
8. Dan Boneh and Matthew K. Franklin. Identity-based encryption from the Weil pairing. pages 213–229, 2001.
9. Xavier Boyen and Brent Waters. Anonymous hierarchical identity-based encryption (without random oracles). pages 290–307, 2006.
10. Ronald Cramer and Victor Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. 33(1):167–226, 2003.
11. Anand Desai. The security of all-or-nothing encryption: Protecting against exhaustive key search. pages 359–375, 2000.
12. Danny Dolev, Cynthia Dwork, and Moni Naor. Non-malleable cryptography, 1998. Manuscript.
13. Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. 31:469–472, 1985.
14. Marc Fischlin. Pseudorandom function tribe ensembles based on one-way permutations: Improvements and applications. pages 432–445, 1999.
15. Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984.
16. Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. 28(4):1364–1396, 1999.
17. Dennis Hofheinz and Enav Weinreb. Searchable encryption with decryption in the standard model. Cryptology ePrint Archive, Report 2008/423, 2008. `http://eprint.iacr.org/`.
18. Jonathan Katz, Amit Sahai, and Brent Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. pages 146–162, 2008.
19. Charles Rackoff and Daniel R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. pages 433–444, 1992.