Improved, Black-Box, Non-Malleable Encryption from Semantic Security

Seung Geol Choi* Dana Dachman-Soled† Tal Malkin‡ Hoeteck Wee§

Abstract

We give a new black-box transformation from any semantically secure encryption scheme into a non-malleable one which has a better rate than the best previous work of Coretti et al. (TCC 2016-A). We achieve a better rate by departing from the "matrix encoding" methodology used by previous constructions, and working directly with a single codeword. We also use a Shamir secret-share packing technique to improve the rate of the underlying error-correcting code.

1 Introduction

Non-malleable encryption. The basic security requirement for public key encryption (PKE) schemes, known as *semantic security* or IND-CPA (indistinguishability under chosen plaintext attack), is that an eavesdropping adversary learns nothing about the plaintext underlying a communicated ciphertext (equivalently, cannot distinguish an encryption of one plaintext from another). Often, however, this indistinguishability guarantee is not sufficient, and a PKE satisfying stronger properties is required.

A strong level of PKE security which is sufficient for most applications, is indistinguishability under chosen-ciphertext attacks (IND-CCA2), wherein the adversary may ask adaptive queries to a decryption oracle (as long as it does not query the "challenge ciphertext" itself). Unfortunately, the various known concrete (cf. [CS03, Wee12, HJ12, LPJY14, KW15, LPJY15, GHKW16]) and generic (cf. [DDN00, CHK04, PW08]) constructions of IND-CCA2 secure encryption either rely on specific number theoretic assumptions, or use seemingly stronger underlying assumptions than IND-CPA secure encryption (e.g., non-interactive zero knowledge, identity-based encryption, or lossy trapdoor functions). Notwithstanding a partial black-box separation result [GMM07], the relationship between IND-CPA and IND-CCA2 security remains unresolved and, therefore, various intermediate notions of security have been proposed and studied.

In this work, we consider the notion of non-malleability under chosen-plaintext attacks (NM-CPA), initially put forward by Dolev, Dwork and Naor [DDN00]. Roughly, non-malleability requires that it is infeasible for an adversary to modify a ciphertext into one, or many, other ciphertexts of messages related to the original plaintext. It was shown by Bellare and Sahai [BS99] and by Pass, shelat and Vaikuntanathan [PSV06] that NM-CPA is equivalent to security against adversaries with access to a non-adaptive decryption oracle, meaning that the adversary can only ask one "parallel"

*United States Naval Academy, choi@usna.edu
†University of Maryland, danadach@ece.umd.edu
‡Columbia University, tal@cs.columbia.edu
§CNRS-DIENS, École Normale Supérieure, wee@di.ens.fr

scheme \ n	$o(k)$	$\Theta(k)$	$\Omega(k^2)$
[CDMW08]	$\Theta(k^3)$	$\Theta(k^2 n)$	$\Theta(k^2 n)$
[CDTV16]	$\Theta(k^3)$	$\Theta(k^2 n)$	$\Theta(kn)$
This work	$\Theta(k^2)$	$\Theta(kn)$	$\Theta(n)$

scheme \ n	$\Theta(k)$	$\Theta(k^2)$	$\Omega(k^3)$
HE + [CDMW08]	$\Theta(k^2 n)$	$\Theta(kn)$	$\Theta(n)$
HE + [CDTV16]	$\Theta(k^2 n)$	$\Theta(kn)$	$\Theta(n)$
HE + this work	$\Theta(kn)$	$\Theta(n)$	$\Theta(n)$

Figure 1: The ciphertext lengths of various NM-CPA encryption schemes. The parameter k is the security parameter, and n is the message length. We assume the underlying IND-CPA encryption has a constant rate for messages of length $\Omega(k)$; encrypting $o(k)$-long messages with IND-CPA encryption is assumed to be $\Theta(k)$-long. HE denotes the hybrid encryption according to Herranz et al. [HHK10]. Note that hybrid encryption doesn't help reduce the ciphertext length for short messages.

CCA2 decryption query. We also consider the notion of non-malleability under bounded-CCA2 attacks (NM-q-CCA2) [CHH$^+$07], where we allow the adversary to adaptively query the decryption oracle at most q times (in contrast, NM-CCA2 allows an *unbounded* number of queries, and is equivalent to IND-CCA2 [DDN00]).

Besides being a "stepping stone" between semantically secure and CCA2 secure encryption, non-malleability (or NM-CPA security) is an important notion in its own right. As one motivating example, consider the use of PKE in auctions. Buyers place their bids for an item to a seller, encrypted under the seller's public key, and the seller sells the item to the buyer with the highest bid. We certainly want to rule out adversaries who consistently bid exactly one dollar more than the previous bidders.

Previous work on achieving NM-CPA from IND-CPA. Interestingly, although NM-CPA appears closer to IND-CCA2 than IND-CPA security, a sequence of results (i.e., a non-black-box construction by [PSV06] followed by a black-box construction by [CDMW08]) showed that NM-CPA schemes (and even NM-q-CCA2 schemes) can be constructed from any IND-CPA scheme.

In a recent work, Coretti et al. [CDTV16] revisited the work of [CDMW08], and investigated (among other results) the question of how *efficient* the black-box transformation can be. The measure of efficiency they consider is the *rate* of the resulting NM-CPA encryption scheme, defined as $\frac{n}{c(n)}$, where n is the message length and $c(n)$ is the length of the corresponding ciphertext.

The transformation of [CDMW08] gives an n-bit NM-CPA scheme such that its encryption algorithm calls the underlying n-bit IND-CPA scheme $\Theta(k^2)$ times, where k is the security parameter.[1] For example, assuming a constant-rate IND-CPA encryption, the transformation gives a $\Theta(k)$-bit NM-CPA scheme with the ciphertext length of $\Theta(k^3)$.

Coretti et al. [CDTV16] give an improved transformation by replacing the error-correcting code used in [CDMW08] with one having a better rate, although the transformation still invokes the same number $\Theta(k^2)$ of calls to the underlying IND-CPA encryption. In particular, this allows $\Theta(k^3)$-bit ciphertexts to encrypt $\Theta(k^2)$-bit messages. See Figure 1 for more detailed comparison.

[1] In fact, according to [CDMW08], the number $\Theta(k^2)$ of calls to IND-CPA encryption can be optimized to $\Theta(k \log^2 k)$; to achieve a negligible soundness error, the scheme checks k random positions, but observe it's enough to check $\log^2 k$ positions since we have $1/2^{\log^2 k} \in neg(k)$. However, we choose to compare the results by using the non-optimized $O(k^2)$ calls, following the presentation of Corretti et al. [CDTV16].

1.1 Our Results

In this work, we give a black-box transformation from IND-CPA encryption to NM-CPA encryption with better efficiency.

Conceptual contribution. Our main conceptual contribution is that *we no longer follow the framework of [DDN00] (and all subsequent constructions) of creating k encryptions of the same message or codeword.*

In particular, as we elaborate on in the next section, previous constructions rely on a "matrix encoding" of the plaintext as a $k \times \ell$ matrix of elements, where each row in the matrix is an encoding of the plaintext message via an appropriate code (the message itself in [DDN00], or more sophisticated encodings in subsequent works). The rows of the matrix are indexed by a one-time signature, so we need at least k (security parameter) rows. It follows that using this methodology incurs a ciphertext expansion of *at least a factor of k, regardless of the underlying code used.* In this sense, [CDTV16] have achieved the best possible rate within this construction framework.

We depart from this "matrix encoding" methodology and work directly with a *single* codeword. This allows us to achieve the first black-box transformation that invokes $\Theta(k)$ calls to the underlying IND-CPA encryption algorithm; previous black-box constructions need $\Theta(k^2)$ calls.

> **Main theorem (informal)** There exists a (fully) black-box construction of a non-malleable encryption scheme from any IND-CPA encryption scheme, in which the encryption algorithm calls the underlying IND-CPA encryption algorithm $\Theta(k)$ times.

We also extend the theorem to provide a black-box construction of NM-q-CCA2 secure encryption [CHH$^+$07] from any semantically secure encryption, calling the IND-CPA encryption algorithm $\Theta(k + q)$ times.

NM-CPA encryption with a better rate. Applying the aforementioned transformation, we achieve an NM-CPA encryption scheme with a better rate. For this, we use a Shamir secret-share packing technique to improve the rate of the underlying error-correcting code to encode the plaintext in the transformation. In particular, we achieve a *constant-rate NM-CPA encryption for messages of length* $\Omega(k^2)$. We compare our results with the previous work in Figure 1.

We note that one can achieve a better rate for long messages by using hybrid encryption. In particular, Herranz et al. [HHK10] showed that NM-CPA KEM plus IND-CCA2 DEM implies NM-CPA PKE. (For shorter messages, the ciphertext length is dominated by the KEM part of encrypting the $\Theta(k)$-long encapsulated key, since for the DEM part, we have a constant-rate IND-CCA2 secure symmetric encryption scheme [BN08].) Even considering the hybrid encryption framework, our scheme achieves better efficiency: Our scheme achieves a constant rate for messages of length $\Omega(k^2)$, rather than for messages of length $\Omega(k^3)$ in the previous schemes.

Potential applications to other related work. The original techniques of [CDMW08] (in particular, the properties of the matrix encoding scheme and its use for verifying consistency) have been used implicitly or explicitly in several works for different purposes. For example, there have been black-box constructions of non-malleable commitments [PW09], set intersection protocols from homomorphic encryptions [DMRY09], and a CCA2-secure encryption scheme for strings starting from one for bits [Ms09]. The works of [Wee10, LP12, KMO14, Kiy14] used these techniques in the context of black-box, round-efficient secure computation. The works of [GLOV12, GOSV14] extended the ideas to provide consistency relations beyond equality using VSS and the paradigm of MPC-in-the-head.

We hope that our improved efficiency, constant rate transformation can be used to improve efficiency in some of these or other application domains. In fact, a very recent work [BDKM16] has already used our results to construct their non-malleble codes resilient against local tampering functions and bounded-depth circuits. Indeed, their results instantiated with the previous matrix encoding techniques would yield non-malleable codes resilient against functions with locality up to n^c for some *specific* $c < 1$ (roughly $c = 1/3$). However, using our results as an ingredient, they were able to achieve resilience against locality $n^{1-\epsilon}$ for *any* constant $\epsilon < 1$ (and even $\frac{n}{\log n}$ with inefficient codes), and much better rate even in lower locality ranges.

2 Techniques

2.1 Overview of Previous Techniques

We begin with an overview of previous techniques of [DDN00, PSV06, CDMW08, CDTV16], which we will refer to below as DDN, PSV, CDMW and CDTV, respectively. We focus on the details that will be helpful towards understanding our techniques.

Non-black-box transformations by DDN and PSV. Let k be the security parameter. The key generation algorithm generates $2k$ independent keys $\textsc{pk}_i^b$ for $i = 1, \ldots, k$, and $b \in \{0, 1\}$ (and the corresponding secret keys). Encryption of message m proceeds as follows:

(a) Generate a $(\textsc{vksig}, \textsc{sksig})$ pair for a one-time signature (where $|\textsc{vksig}| = k$).

(b) Generate k encryptions of the message m. In particular, use keys $\textsc{pk}_i^{\textsc{vksig}_i}$ for $i = 1, \ldots, k$ for encryptions.

(c) Give a non-interactive zero-knowledge proof (or the relaxed "designated verifier" version) proving that all resulting ciphertexts are encryptions of the same message.

(d) Sign the entire bundle with a one-time signature.

It is in step (c) that a general NP-reduction is used, which in return makes the construction non-black-box (and inefficient). In the proof of security, we exploit that fact that for a well-formed ciphertext, we can recover the message if we know the secret key for *any* of the k encryptions.

Black-box transformations by CDMW. Let k be the security parameter, and let $\ell = O(k)$ (or any superlogarithmic function in k). The key generation algorithm generates $2k\ell$ independent keys $\textsc{pk}_{i,j}^b$ for $i = 1, \ldots, k$, $j = 1, \ldots, \ell$, and $b \in \{0, 1\}$ (and the corresponding secret keys). The encryption algorithm utilizes a Reed-Solomon error correcting code with encoding algorithm E. Now, the encryption algorithm has the following form:

(a) Generate a $(\textsc{vksig}, \textsc{sksig})$ pair for a one-time signature (where $|\textsc{vksig}| = k$).

(b) Obtain an encoding w of a message m by computing $w \leftarrow \mathsf{E}(m)$. Generate k encryptions of the same codeword w, using ℓ public keys per each of the k encryptions in a way that we explain below.

(d) Sign the entire bundle with a one-time signature.

Obviously, the scheme should provide *some mechanism for checking the consistency of k encryptions*, corresponding to step (c) in DDN and PSV (i.e., the non-interactive zero-knowledge proof). That way, even if the simulated decryption in the proof of security decrypts any of the k ciphertexts, the decryption should be correct with overwhelming probability. CDMW achieved this by using a w consisting of ℓ elements, and encrypting each element with a different public key, for a total of $k\ell$ encryptions. The decryption algorithm checks consistency of the k encryptions of w *by checking consistency of a random subset of columns* (where the randomness is determined by its secret key). Then, the decryption algorithm decrypts and error-corrects the first row, and checks that in that same subset of locations, this codeword is not corrupted. If both these column-check and codeword-check pass, output the decoded message.

We next describe the details of how the above outline is implemented, and the intuition behind its security and parameter choices. Recall that a Reed-Solomon (RS) codeword consisting of ℓ output symbols is simply a polynomial p of degree $d = O(\ell)$ over a finite field, evaluated at ℓ points (say $1, \dots, \ell$). The way CDMW encode a message m is via Shamir secret-sharing, which can be viewed as an instantiation of a RS code. Specifically, set $p(0) = m$, choose the values of $p(1), \dots, p(d)$ at random, and interpolate to obtain the unique degree d polynomial p. Let the final encoding $w \leftarrow \mathsf{E}(m)$ consist of $w_1 = p(1), \dots, w_\ell = p(\ell)$. The encryptions now proceed as follows:

> Construct a $k \times \ell$ matrix M, where $M_{i,j} = w_j$ and k is the number of bits in $\textsc{vksig} = \textsc{vksig}_1, \dots, \textsc{vksig}_k$. Each entry of $M_{i,j}$ is then encrypted under a one of two public keys $(\textsc{pk}^0_{i,j}, \textsc{pk}^1_{i,j})$, depending on whether $\textsc{vksig}_i$ is 0 or 1.

In the actual decryption algorithm, the first row of the encrypted matrix is always decrypted and decoded, whereas in the security proof, the decrypted row will be chosen based on which secret keys are available to the reduction, and it is ensured that in each submitted ciphertext there is some row for which the reduction knows all the secret keys. The key challenge is to ensure that decrypting and decoding *any one* of the k rows of the encrypted matrix will yield *the same* message $\widetilde{m}$ (possibly $\perp$) as the decrypting and decoding the first row. This is where the "column check" and "codeword check" come in. In the column check, we decrypt a random subset of $t = O(\ell)$ columns, and check that all the entries in each of these columns are the same; the random subset is chosen in key generation and embedded into the private key. Intuitively, this ensures that the encoding in each row is "close" to the encoding in the first row. In the codeword check, we decrypt and decode the first row and then check the resulting codeword against the received word in the first row. Specifically, we check that $t = O(\ell)$ random positions of the first row (the same ones that were opened during the column check) agree with the corresponding t positions in the decoded codeword. Intuitively, this is a type of a cut-and-choose check which ensures that the encoding in the first row is "close" to a valid codeword. If either of the checks fails, we output $\perp$. Put together, the two checks ensure that with overwhelming probability, all rows must decode to the same message (or to $\perp$), and thus provide the desired consistency.

The reason CDMW needs ℓ to be superlogarithmic, is that for the codeword check, we need the number of random positions $t = O(\ell)$ to satisfy $2^{-t} = \mathsf{negl}(k)$ so that a codeword that is far from valid will pass the check with negligible probability. Thus, the RS code used for each row is not constant rate.

More efficient black-box transformation by CDTV. The general insight of CDTV (and also the full version of CDMW [CDMW16]) is that the above construction can be generalized to work for a larger class of encoding schemes E, beyond just Reed-Solomon codes. Specifically, Coretti et al. [CDTV16] note that using a LECSS (linear error-correcting secret sharing scheme) is sufficient [CG14, CDD+15], whereas [CDMW16] introduced a notion of reconstructable probabilistic encoding scheme (building on [DGR99]). Using these insights, the above works were able to replace the RS code described above with a constant-rate encoding scheme (for long enough messages). Specifically, each row with ℓ elements can in fact encode a message of length $O(\ell)$ elements, resulting in a constant rate code for each row (while still maintaining k rows).

2.2 Our Techniques

Our encryption scheme also utilizes reconstructable probabilistic encoding (RPE) schemes. RPE schemes are, informally, error-correcting codes with additional secrecy and reconstruction properties. The secrecy property guarantees that the symbols at any not-too-large subset of positions in the codeword are distributed uniformly and independently of the encoded message. The reconstruction property says that furthermore, any assignment of symbols to such a subset of positions, can be completed to a (correctly distributed) codeword for any given message. The parameter regime we will be interested in is the standard one, where the error-correction is with respect to a constant fraction of errors, and the secrecy and reconstruction are also with respect to a (smaller) constant fraction of positions.

From k encryptions to a single encryption. Our first technical contribution is identifying a property of RPE schemes and showing how it can be leveraged to eliminate the need for the "repetition" encoding in previous works. The property we use is that error-correction and decoding can be performed given *any* large enough (constant-fraction) sized subset $\lambda \cdot \ell$ of positions of the corrupted codeword (here $0 < \lambda < 1$ is a constant). We call this property the "decoding from partial views" property. Crucially, we would like this property to hold in a strong way, so that for *any* such subset, we always decode to *the same* codeword/message (possibly $\perp$), even for arbitrarily corrupt codewords, with overwhelming probability (taken over the random choice of the secret key).

We have already discussed a similar property as underlying, at least implicitly, the previous works relying on matrix encoding. However, in those works the property applied to decoding from any one of the k rows (which constitute a repetition code), and was unrelated to the use of RPE for the encoding within each row. Our novel observation is that in fact a similar property can apply *directly* to RPE (with appropriate parameters). A *single* RPE codeword could then allow decoding from any partial view subset, and by correctly adapting a codeword check and another layer of (standard) encoding on the signature, we can achieve the strong version guaranteeing consistency with overwhelming probability.

Thus, encryption of a message m proceeds as follows:

(a) Generate a $(\textsc{vksig}, \textsc{sksig})$ pair for a one-time signature.

(b) Let E be the encoding algorithm of a RPE with the output length ℓ. Let C be a linear code with relative distance $\lambda < 1$, encoding the length-k $\textsc{vksig}$ to a length-ℓ string (note that C does not have to be efficiently error correctable). Set $\vec{s} \leftarrow \mathsf{E}(m)$, where $\vec{s}$ is a vector of length ℓ. Let $v_1 \cdots v_\ell$ be the output of $C(\textsc{vksig})$. For $j = 1, \ldots, \ell$, encrypt each entry s_j under public key $\textsc{pk}_j^{v_j}$, yielding a vector of ciphertexts.

(c) Sign the entire bundle with a one-time signature.

In the actual decryption algorithm, the first $\lambda \cdot \ell$ positions of the ciphertext vector are always decrypted and decoded, whereas in the security proof, a specific subset of size $\lambda \cdot \ell$ will be chosen based on which secret keys are available to the reduction. In the proof, we use the fact that, due to unforgeability of the signature scheme, the VKSIG for each submitted ciphertext must be different than the VKSIG of the challenge ciphertext, and the fact that C has distance $\lambda \cdot \ell$ to ensure that there is always some sufficiently large subset for which the reduction knows all the secret keys.

To ensure that decoding any of subset of size $\lambda \cdot \ell$ positions yields the same message $\widetilde{m}$ as the first subset (or both will give $\perp$), we require an analogue of the codeword check only (but no column check). As before, in the codeword check we compare the codeword obtained by decrypting and decoding the first $\lambda \cdot \ell$ positions with the received word. Specifically, we check that t random positions throughout the entire received word agree with the corresponding t positions in the decoded codeword.

Constant-rate RPE. Our second contribution is to show that the above framework is implementable with a constant rate RPE: we show that using Reed-Solomon codes or packed Shamir secret sharing yields a constant rate RPE with appropriate parameters. Compared with the Reed-Solomon based encodings used by [CDMW08] for k rows, here our encoding has a single row (of length a constant times larger), and a longer message is encoded in each codeword via the packing technique [FY92]. That is, the polynomial is taken to be of a larger degree, and the message is encoded in several evaluation points of the polynomial.

3 Preliminaries and Definitions

We use $[n]$ to denote $\{1, 2, \ldots, n\}$. If A is a probabilistic polynomial time (hereafter, ppt) algorithm that runs on input x, $A(x)$ denotes the random variable according to the distribution of the output of A on input x. We denote by $A(x; r)$ the output of A on input x and random coins r. Computational indistinguishability between two ensembles A and B is denoted by $A \stackrel{c}{\approx} B$, and statistical indistinguishability between two distributions A and B is denoted by $A \stackrel{s}{\approx} B$.

Distance of two strings. Given two strings v, w of length ℓ over an alphabet Σ, we say that v and w are δ-far if they disagree in more than $\delta \cdot \ell$ positions, where $0 \le \delta \le 1$; we say that v and w are δ-close if they agree in more than $\delta \cdot \ell$ positions.

3.1 Semantically Secure Encryption

Definition 1 (Encryption scheme)**.** *A triple* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is an* encryption scheme*, if* Gen *and* Enc *are ppt algorithms and* Dec *is a deterministic polynomial-time algorithm which satisfies the following property:*

> Correctness. *There exists a negligible function* $\mu(\cdot)$ *such that for all sufficiently large* k*, we have that with probability* $1-\mu(k)$ *over* $(\text{PK}, \text{SK}) \leftarrow \mathsf{Gen}(1^k)$*: for all* m*,* $\Pr[\mathsf{Dec}_{\text{SK}}(\mathsf{Enc}_{\text{PK}}(m)) = m] = 1$*.*

Definition 2 (Semantic security)**.** *Let* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be an encryption scheme and let the random variable* $\mathsf{IND}_b(\Pi, A, k)$*, where* $b \in \{0,1\}$*,* $A = (A_1, A_2)$ *are ppt algorithms and* $k \in \mathbb{N}$*, denote the result of the following probabilistic experiment:*

$\mathsf{IND}_b(\Pi, A, k)$:
  $(\text{PK}, \text{SK}) \leftarrow \mathsf{Gen}(1^k)$
  $(m_0, m_1, \text{STATE}_A) \leftarrow A_1(\text{PK})$ *s.t.* $|m_0| = |m_1|$
  $y \leftarrow \mathsf{Enc}_{\text{PK}}(m_b)$
  $D \leftarrow A_2(y, \text{STATE}_A)$
  Output D

$(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is* indistinguishable under a chosen-plaintext (CPA) attack*, or* semantically secure*, if for any ppt algorithms* $A = (A_1, A_2)$ *the following two ensembles are computationally indistinguishable:*

$$\left\{\mathsf{IND}_0(\Pi, A, k)\right\}_{k\in\mathbb{N}} \stackrel{c}{\approx} \left\{\mathsf{IND}_1(\Pi, A, k)\right\}_{k\in\mathbb{N}}$$

It follows from a straight-forward hybrid argument that semantic security implies indistinguishability of multiple encryptions under independently chosen keys:

Proposition 1. *Let* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be a semantically secure encryption scheme and let the random variable* $\mathsf{mIND}_b(\Pi, A, k, \ell)$*, where* $b \in \{0,1\}$*,* $A = (A_1, A_2)$ *are ppt algorithms and* $k \in \mathbb{N}$*, denote the result of the following probabilistic experiment:*

$\mathsf{mIND}_b(\Pi, A, k, \ell)$:
  For $i = 1, \ldots, \ell$*:* $(\text{PK}_i, \text{SK}_i) \leftarrow \mathsf{Gen}(1^k)$
  $(\langle m_0^1, \ldots, m_0^\ell\rangle, \langle m_1^1, \ldots, m_1^\ell\rangle, \text{STATE}_A) \leftarrow A_1(\langle \text{PK}_1, \ldots, \text{PK}_\ell\rangle)$
    s.t. $|m_0^1| = |m_1^1| = \cdots = |m_0^\ell| = |m_1^\ell|$
  For $i = 1, \ldots, \ell$*:* $y_i \leftarrow \mathsf{Enc}_{\text{PK}_i}(m_b^i)$
  $D \leftarrow A_2(y_1, \ldots, y_\ell, \text{STATE}_A)$
  Output D

then for any ppt algorithms $A = (A_1, A_2)$ *and for any polynomial* $p(k)$ *the following two ensembles are computationally indistinguishable:*

$$\left\{\mathsf{mIND}_0(\Pi, A, k, p(k))\right\}_{k\in N} \stackrel{c}{\approx} \left\{\mathsf{mIND}_1(\Pi, A, k, p(k))\right\}_{k\in N}$$

3.2 Non-malleable Encryption

Definition 3 (Non-malleable encryption [PSV06])**.** *Let* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *be an encryption scheme and let the random variable* $\mathsf{NME}_b(\Pi, A, k, \ell)$ *where* $b \in \{0,1\}$*,* $A = (A_1, A_2)$ *are ppt algorithms and* $k, \ell \in \mathbb{N}$ *denote the result of the following probabilistic experiment:*

$\mathsf{NME}_b(\Pi, A, k, \ell)$:
  $(\text{PK}, \text{SK}) \leftarrow \mathsf{Gen}(1^k)$
  $(m_0, m_1, \text{STATE}_A) \leftarrow A_1(\text{PK})$ *s.t.* $|m_0| = |m_1|$
  $y \leftarrow \mathsf{Enc}_{\text{PK}}(m_b)$
  $(\psi_1, \ldots, \psi_\ell) \leftarrow A_2(y, \text{STATE}_A)$
  Output $(d_1, \ldots, d_\ell)$ *where* $d_i = \begin{cases} \perp & \text{if } \psi_i = y \\ \mathsf{Dec}_{\text{SK}}(\psi_i) & \text{otherwise} \end{cases}$

$(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is* non-malleable under a chosen plaintext (CPA) attack *if for any ppt algorithms $A = (A_1, A_2)$ and for any polynomial $p(k)$, the following two ensembles are computationally indistinguishable:*

$$\left\{\mathsf{NME}_0(\Pi, A, k, p(k))\right\}_{k\in\mathbb{N}} \stackrel{c}{\approx} \left\{\mathsf{NME}_1(\Pi, A, k, p(k))\right\}_{k\in\mathbb{N}}$$

It was shown in [PSV06] that an encryption that is non-malleable (under Definition 3) remains non-malleable even if the adversary A_2 receives several encryptions under many different public keys (the formal experiment is the analogue of mIND for non-malleability).

3.3 Bounded-CCA2 Non-Malleability

The definition of Bounded-CCA2 Non-Malleability is almost identical to the definition of Non-Malleability except here, we allow the adversary to query Dec at most q times in the non-malleability experiment (but it must not query Dec on the challenge ciphertext).

Definition 4 (Bounded-CCA2 non-malleable encryption [CHH+07])**.** *Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be an encryption scheme and let the random variable $\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b(\Pi, A, k, \ell)$ where $b \in \{0,1\}$, $A = (A_1, A_2)$ are ppt algorithms and $k, \ell \in \mathbb{N}$ denote the result of the following probabilistic experiment:*

$$\begin{array}{l}
\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b(\Pi, A, k, \ell): \\
\quad (\text{PK}, \text{SK}) \leftarrow \mathsf{Gen}(1^k) \\
\quad (m_0, m_1, \text{STATE}_A) \leftarrow A_1^{O_1}(\text{PK}) \ \ s.t.\ |m_0| = |m_1| \\
\quad y \leftarrow \mathsf{Enc}_{\text{PK}}(m_b) \\
\quad (\psi_1, \ldots, \psi_\ell) \leftarrow A_2^{O_2}(y, \text{STATE}_A) \\
\quad \textit{Output } (d_1, \ldots, d_\ell) \textit{ where } d_i = \begin{cases} \perp & \textit{if } \psi_i = y \\ \mathsf{Dec}_{\text{SK}}(\psi_i) & \textit{otherwise} \end{cases}
\end{array}$$

$(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is* non-malleable under a bounded-CCA2 attack *for a function $q(k) : \mathbb{N} \to \mathbb{N}$ if $\forall$ ppt algorithms $A = (A_1, A_2)$ which make $q(k)$ total queries to the oracles and for any polynomial $p(k)$, the following two ensembles are computationally indistinguishable:*

$$\left\{\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_0(\Pi, A, k, p(k))\right\}_{k\in\mathbb{N}} \stackrel{c}{\approx} \left\{\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_1(\Pi, A, k, p(k))\right\}_{k\in\mathbb{N}}$$

The oracle $O_1 = \mathsf{Dec}_{\text{SK}}(\cdot)$ is the decryption oracle. $O_2 = \mathsf{Dec}^y_{\text{SK}}(\cdot)$ is the decryption oracle except that O_2 returns $\perp$ when queried on y.

3.4 (Strong) One-Time Signature Schemes

Informally, a (strong) one-time signature scheme $(\mathsf{GenSig}, \mathsf{Sign}, \mathsf{VerSig})$ is an existentially unforgeable signature scheme, with the restriction that the signer signs at most one message with any key. This means that an efficient adversary, upon seeing a signature on a message m of his choice, cannot generate a valid signature on a different message, or a different valid signature on the same message m. Such schemes can be constructed in a black-box way from one-way functions [Rom90, Lam79], and thus from any semantically secure encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ using black-box access only to Gen.

In this paper, signature/verification key pairs are sometimes represented as strings over a non-binary alphabet; this technique has also been used in [RS10]. This augmented version can simply be cast into the version over the binary alphabet, by trivially encoding such keys into a binary string.

4 Reconstructable Probabilistic Encoding Scheme (RPE)

4.1 RPE Definition

We assume that the readers are familiar with basic notions of error correcting codes, including linear codes and Reed-Solomon codes. We define a reconstructable probabilistic encoding (RPE) below. The secrecy property of an RPE implies that short partial codewords are not bound to any message. The reconstruction property implies that one can later bind such a short partial codeword to any target message and create the whole consistent codeword. Jumping ahead, this reconstruction procedure will be used to create two different messages sharing the same partial codeword in the reduction step of the proof.

There are several parameters in RPE schemes. A message is represented as a binary string, and the parameter n specifies the length of a message (in bits). A codeword is represented as a string over alphabet Σ, and the parameter ℓ specifies the length of a codeword in the codeword space. The parameter δ is used to specify the relative distance between codewords. The parameter t is used to specify a threshold to determine whether a partial codeword is short; every codeword of length at most t is considered short.

Definition 5 (Reconstructable probabilistic encoding)**.** *We say a triple* $(\mathsf{E}, \mathsf{D}, \mathsf{R})$ *is a* reconstructable probabilistic encoding scheme *with parameters* $(n, \ell, \delta, t, \Sigma)$*, where* $n, \ell, t \in \mathbb{N}$*,* $0 < \delta < 1$*, and* Σ *is an alphabet, if it satisfies the following properties:*

1. ***Error correction.*** $\mathsf{E} : \{0,1\}^n \to \Sigma^\ell$ *is an efficient probabilistic procedure, which maps a message* $m \in \{0,1\}^n$ *to a distribution over* Σ^ℓ*. If we let* $\mathcal{W}$ *denote the support of* E*, any two strings in* $\mathcal{W}$ *are* δ*-far. Moreover,* D *is an efficient procedure that given any* $w' \in \Sigma^\ell$ *that is* $(1 - \delta/2)$*-close to some string* w *in* $\mathcal{W}$*, outputs* w *along with a consistent* m*.*
2. ***Secrecy of short partial codewords.*** *For all* $m \in \{0,1\}^n$ *and all sets* $S \subset [\ell]$ *of size* t*, the projection of* $\mathsf{E}(m)$ *onto the coordinates in* S*, as denoted by* $\mathsf{E}(m)|_S$*, is identically distributed to the uniform distribution over* Σ^t*.*
3. ***Reconstruction from short partial codewords.*** R *is an efficient procedure that given any set* $S \subset [\ell]$ *of size* t*, any* $(\alpha_1, \ldots, \alpha_t) \in \Sigma^t$*, and any* $m \in \{0,1\}^n$*, samples from the distribution* $\mathsf{E}(m)$ *with the constraint* $\mathsf{E}(m)|_S = (\alpha_1, \ldots, \alpha_t)$*.*

We note that similar properties have been exploited already in the early work on secure multiparty computation of Ben-Or, Goldwasser, and Wigderson [BGW88], with encoding via low-degree polynomials (or Reed-Solomon codes or Shamir secret sharing with Berlekamp-Welch correction). The above notion of RPE was explicitly defined in [CDMW16], extending a definition given by Decatur, Goldreich, and Ron [DGR99], who only required error-correction and secrecy, but not reconstruction.

4.2 Decoding from Partial Views

The following property will be useful for our construction of non-malleable encryption. Informally, the following lemma states that for any RPE as above, given a sufficiently large "partial view" (i.e. subset of positions), decoding with error correction can be successfully performed on this partial view.

Lemma 1. *Let* $(\mathsf{E}, \mathsf{D}, \mathsf{R})$ *be a* reconstructable probabilistic encoding scheme *with parameters* $(n, \ell, \delta, t, \Sigma)$ *then*

> *Let* $\lambda := (1 - \delta/4)$, $\delta' := \delta/2$. *For any set* $S \in [\ell]$ *with size* $s := \lambda \cdot \ell$ *and given any* $w' \in \Sigma^s$ *that is* $(1 - \delta'/2)$*-close to* $w|_S$ *for some string* w *in* $\mathcal{W}$*, there is an efficient procedure* $\mathsf{D}'(S, w')$ *that outputs* w *along with a consistent* m*.*

Proof. Let $S = \{i_1, \ldots, i_s\}$, where $s = \lambda \cdot \ell$. The decoding from partial views procedure $\mathsf{D}'(S, w')$ does the following. Define the string $\widetilde{w} = \widetilde{w}_1, \ldots, \widetilde{w}_\ell \in \Sigma^\ell$ in the following way: For $j \in [s]$, $\widetilde{w}_{i_j} := w'_j$ and for $v \in [\ell] \setminus S$, $\widetilde{w}_v = \sigma$, where σ is an arbitrary symbol in Σ. Note that if w' is $(1 - \delta'/2)$-close to $w|_S$ for some string w in $\mathcal{W}$, then $\widetilde{w}$ is $(1 - \delta/2)$-close to w for some string w in $\mathcal{W}$. Therefore, running the regular decode procedure, $\mathsf{D}(\widetilde{w})$ is guaranteed to output (w, m), where $w \in \mathcal{W}$ is the corrected codeword, and m is the original message. □

4.3 RPE from Reed-Solomon Codes

In this section, we construct such a constant-rate RPE scheme with a RS (Reed-Solomon) code and packed secret-sharing [FY92]. We note this is a simple construction; similar constructions were given in different contexts.

Construction 1. (RS-based RPE)
For any $n, t, \gamma \in \mathbb{N}$, and for any δ with $0 < \delta < 1$, we construct an RPE scheme with parameters $(n, \ell, \delta, t, \Sigma)$ where $\ell = \lceil \frac{t+u}{1-\delta} \rceil$ with $u = \lceil n/\gamma \rceil$ and $\Sigma = \mathrm{GF}(2^\gamma)$.

We implicitly associate a string $m \in \{0,1\}^n$ with a vector $(m_1, m_2, \ldots, m_u)$ where each $m_i \in \mathrm{GF}(2^\gamma)$; an integer i with $0 \le i < 2^\gamma$ will also be implicitly encoded into a field element in $\mathrm{GF}(2^\gamma)$. We construct an RPE scheme $(\mathsf{E}, \mathsf{D}, \mathsf{R})$ as follows:

- $\mathsf{E}(m)$: Let $d = t + u - 1$. Choose a random degree-d polynomial q over Σ such that $q(\ell+1) = m_1, \ldots, q(\ell+u) = m_u$ and output $w = (q(1), q(2), \ldots, q(\ell))$.
- $\mathsf{D}(w')$: Decode w' using the Berlekamp-Welch algorithm and output (w, m), where w is the corrected codeword, and m is the original message.
- $\mathsf{R}(S, (\alpha_1, \ldots, \alpha_t), m)$: Let $S = \{i_1, \ldots, i_t\}$. Determine the degree-d polynomial q such that $q(i_1) = \alpha_1, \ldots, q(i_t) = \alpha_t$ and $q(\ell+1) = m_1, \ldots, q(\ell+u) = m_u$. Output $(q(1), \ldots, q(\ell))$.

Error correction property holds since we simply use the Reed-Solomon code $\mathcal{W}$ in encoding and decoding, where

$$\mathcal{W} = \{\ (q(1), \ldots, q(\ell)) \mid q \text{ is a degree-}d \text{ polynomial }\}.$$

Note that $\mathcal{W}$ is a code over the alphabet Σ with minimum relative distance is $\frac{\ell-d+1}{\ell} > \delta$, which means we may efficiently correct up to $\delta/2$ fraction errors. Secrecy and reconstruction properties hold since the codeword $(q(1), \ldots, q(\ell))$ is a $(t+u)$-out-of-ℓ secret-sharing of m using Shamir's secret-sharing scheme, and $(\alpha_1, \ldots, \alpha_t, m_1, \ldots, m_u)$ allows the reconstruction of the (one and only) degree-d polynomial.

Decoding from partial views with better parameters. By using the property of Reed-Solomon codes, we can obtain a better parameters in terms of decoding from partial views.

Lemma 2. *Let* $(\mathsf{E}, \mathsf{D}, \mathsf{R})$ *be an RPE scheme with parameters* $(n, \ell, \delta, t, \Sigma)$*, according to Construction 1. Then for any set* $V \subseteq [\ell]$ *with size* $s = \lambda\ell$ *such that* $t + u < s \leq \ell$ *and for any* $w' \in \Sigma^s$ *that is* $(1 - \delta'/2)$*-close to* $w|_V$ *for some string* w *in* $\mathcal{W}$*, where* $\delta' = \frac{s-(t+u-1)}{s}$*, there is an efficient procedure* $\mathsf{D}'(V, w')$ *that outputs* w *along with a consistent* m*.*

Proof. Let $V = \{i_1, \ldots, i_s\}$. Then,

$$\mathcal{W}' = \{\ (q(i_1), \ldots, q(i_s)) \mid q \text{ is a degree-}d \text{ polynomial }\}$$

forms another Reed-Solomon code, where $d = t + u - 1$. Note that $\mathcal{W}'$ is a code over the alphabet $\mathrm{GF}(2^\gamma)$ with minimum relative distance is $\frac{s-d+1}{s} > \delta'$, which means we can decode w' using the Berlekamp-Welch algorithm, correcting up to $\delta'/2$ fraction errors. □

Example instantiation 1. By applying Construction 1 with $\delta = 0.9$ and $\gamma = n$, we obtain an RPE with parameters $(n, 10t, 0.9, t, \mathrm{GF}(2^n))$. According to Lemma 2 with $\lambda = 0.3$, the scheme can decode a partial codeword of length $3t$, correcting up to $\delta'/2 = 1/3$ fraction errors.

Example instantiation 2 (constant-rate RPE). By applying Construction 1 with $\delta = 0.9$ and $\gamma = n/t$, we obtain an RPE with parameters $(n, 20t, 0.9, t, \mathrm{GF}(2^\gamma))$ with rate 0.05. According to Lemma 2 with $\lambda = 0.3$, the scheme can decode a partial codeword of length $6t$, correcting up to $\delta'/2 = 1/3$ fraction errors.

5 Non-malleable Encryption from Semantically Secure One

5.1 Generic Construction Using Any RPE

Given a semantically secure encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ and a RPE, we construct a non-malleable encryption scheme $\Pi = (\mathsf{NMGen}^{\mathsf{Gen}}, \mathsf{NMEnc}^{\mathsf{Gen},\mathsf{Enc}}, \mathsf{NMDec}^{\mathsf{Gen},\mathsf{Dec}})$, summarized in Figure 2 and described as follows.

Key generation. Let k be the security parameter. Let $(\mathsf{E}, \mathsf{D}, \mathsf{R})$ be an RPE scheme with parameters $(n, \ell, \delta, t, \Sigma)$ and let λ and δ' be the parameter associated with decoding the partial views. In addition, set $t = k$.

The public key contains an error correcting code $C : \Gamma^t \to \Gamma^\ell$ with the distance $\lambda\ell$, where Γ is an appropriately chosen finite alphabet in order to satisfy the distance condition. Let $g = |\Gamma|$ and we will implicitly associated $\Gamma = [g]$. We note this technique was used in [RS10]. In addition, there are $g \cdot \ell$ public keys from Gen indexed by a triplet $(j, b) \in [\ell] \times [g]$, that is, $\{\mathrm{PK}_j^b \mid (j, b) \in [\ell] \times [g]\}$.

The secret key contains the decryption keys SK_j^bs and a random subset S of $[\ell]$ with size t to be used in decryption for consistency checks (described below).

Encryption. Encryption of a message $m \in \{0,1\}^n$ proceeds as follows:

1. Generate $(\textsc{sksig}, \textsc{vksig})$ for a one-time signature where $\textsc{vksig} \in \Gamma^t$, and compute $(v_1, \ldots, v_\ell) \leftarrow C(\textsc{vksig})$.

Let $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be an encryption scheme, $(\mathsf{GenSig}, \mathsf{Sign}, \mathsf{VerSig})$ be a strong one-time signature scheme, and $(\mathsf{E}, \mathsf{D}, \mathsf{R})$ be a reconstructable probabilistic encoding scheme with parameters $(n, \ell, \delta, t, \Sigma)$. Moreover, let λ and δ' be the parameters associated with decoding the partial views.

Setting the RPE parameter t: To achieve NM-CPA, set $t = k$ and to achieve NM-q-CCA2, set $t = a(k + q(k))$, where a is a constant such that $\left(1 - \frac{\lambda\delta'}{2}\right)^a \leq \frac{1}{2}$.

$\mathsf{NMGen}(1^k)$:

1. Choose an error correcting code $C : \Gamma^t \to \Gamma^\ell$ with the distance $\lambda\ell$, where Γ is an appropriately chosen finite alphabet to satisfy the distance condition. Let $g = |\Gamma|$, and we will implicitly associate Γ with $[g]$.
2. For $j \in [\ell], b \in [g]$, run $\mathsf{Gen}(1^k)$ to generate key-pairs $(\textsc{pk}_j^b, \textsc{sk}_j^b)$.
3. Pick a random subset $S \subset [\ell]$ of size t.
4. Set $\textsc{pk} = \left(C, \{\textsc{pk}_j^b \mid j \in [\ell], b \in [g]\}\right)$ and $\textsc{sk} = \left(S, \{\textsc{sk}_j^b \mid j \in [\lambda\ell] \cup S, b \in [g]\}\right)$.

$\mathsf{NMEnc}_{\textsc{pk}}(m)$:

1. Run $\mathsf{GenSig}(1^k)$ to generate $(\textsc{sksig}, \textsc{vksig})$. Parse $\textsc{vksig}$ as an element $\textsc{vksig} \in \Gamma^t$. Let $C(\textsc{vksig}) := (v_1, \ldots, v_\ell)$, where $v_1, \ldots, v_\ell \in \Gamma^\ell$.
2. Compute $(s_1, \ldots, s_\ell) \leftarrow \mathsf{E}(m)$, where $m \in \{0,1\}^n$. Compute the ciphertext $c_j \leftarrow \mathsf{Enc}_{\textsc{pk}_j^{v_j}}(s_j)$, for $j \in [\ell]$.
3. Compute the signature $\sigma \leftarrow \mathsf{Sign}_{\textsc{sksig}}(\vec{c})$ where $\vec{c} = (c_1, \ldots, c_\ell)$.
4. Output the tuple $[\vec{c}, \textsc{vksig}, \sigma]$.

$\mathsf{NMDec}_{\textsc{sk}}([\vec{c}, \textsc{vksig}, \sigma])$:

1. (`sig-check`) Verify the signature with $\mathsf{VerSig}_{\textsc{vksig}}[\vec{c}, \sigma]$.
2. (`decoding-check`) Let $\vec{c} = (c_j)$ and $(v_1, \ldots, v_\ell) = C(\textsc{vksig})$. For $j \in [\lambda\ell]$, compute $s_j = \mathsf{Dec}_{\textsc{sk}_j^{v_j}}(c_j)$. Compute $((w_1, \ldots, w_\ell), m) \leftarrow \mathsf{D}'([\lambda\ell], (s_1, \ldots, s_{\lambda\ell}))$. If the decoding fails or $(w_1, \ldots, w_{\lambda\ell})$ is $\frac{\delta'}{2}$-far from $(s_1, \ldots, s_{\lambda\ell})$, then output $\perp$.
3. (`codeword-check`) Compute $s_j = \mathsf{Dec}_{sk_j^{v_j}}(c_j)$ for all $j \in S$. Check that $s_j = w_j$.
4. If all the checks accept, output the message m corresponding to the codeword w; else, output $\perp$.

Figure 2: The Non-Malleable Encryption Scheme Π

2. Compute $(s_1, \ldots, s_\ell) \leftarrow \mathsf{E}(m)$ and compute ℓ-long vector $\vec{c} = (c_1, \ldots, c_\ell)$ of ciphertexts where $c_j = \mathsf{Enc}_{\mathrm{PK}_j^{v_j}}(s_j)$:

$$\vec{c} = \left(\mathsf{Enc}_{\mathrm{PK}_1^{v_1}}(s_1), \mathsf{Enc}_{\mathrm{PK}_2^{v_2}}(s_2), \ldots, \mathsf{Enc}_{\mathrm{PK}_\ell^{v_\ell}}(s_\ell)\right)$$

3. Create a signature σ on $\vec{c}$ using SKSIG. The ciphertext is $[\text{VKSIG}, \vec{c}, \sigma]$.

Decryption. To decrypt, we verify the signature and perform consistency checks. A valid ciphertext in Π is an encryption of a codeword in $\mathcal{W}$. We want to design consistency checks that reject ciphertexts that are "far" from being valid ciphertexts under Π. For simplicity, we will describe the consistency checks as applied to the underlying vector of plaintexts. The checks depend on a random subset S of t columns chosen during key generation.

`decoding-check`: Let $I = \{1, \ldots, \lambda\ell\}$. We find a codeword w such that $w|_I$ is $(1 - \frac{\delta'}{2})$-close to the first $\lambda\ell$ elements of the vector $(s_1, \ldots, s_\ell)$; the check fails if no such w exists. Recall that according to Lemma 1, it can correct up to $\delta'/2$ fraction errors of $(s_1, \ldots, s_{\lambda\ell})$.

`codeword-check`: We check that the vector $(s_1, \ldots, s_\ell)$ agrees with w at the positions indexed by S.

Finally, if all the checks accept, decode the codeword w and output the result; otherwise output $\perp$.

We note that we only need a partial set of the decryption keys, in particular for I and S, in order to achieve the decryption procedure.

5.2 Using Construction 1 for RPE.

By plugging RS-based RPE in Construction 1, using parameters $(n, 20k, 0.9, k, \mathrm{GF}(2^{n/k}))$, into the above generic NM-CPA construction, we obtain an NM-CPA encryption scheme for messages of length $\omega(k)$. The underlying IND-CPA scheme encrypts an element of $\mathrm{GF}(2^{n/k})$, and there are $20k$ of them in the overall NM-CPA ciphertext; if the underlying IND-CPA encryption is constant rate, the overall NM-CPA also achieves a constant rate. We give a self-contained description when $n = \omega(k)$ in Figure 3. Note *we can use the binary alphabet for the error correcting code C, since it has a small relative distance of* $\lambda = 0.3$.

When the message is of length $O(k)$, we can instantiate an NM-CPA encryption scheme by using an RPE with parameters $(n, 10k, 0.9, k, \mathrm{GF}(2^n))$. Note the overall NM-CPA ciphertext length becomes $\Theta(k^2)$. The underlying IND-CPA scheme encrypts an element of $\mathrm{GF}(2^n)$, and there are $10k$ of them in the overall NM-CPA ciphertext.

6 Analysis

Theorem 1. *If* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is a semantically secure PKE, then the PKE scheme* Π *described in Figure 2 is non-malleable under a chosen plaintext attack.*

6.1 Proof of Main Theorem

In the hybrid argument, we consider the following variants of NME_b as applied to Π, where VKSIG^* denotes the verification key in the ciphertext $y = \mathsf{NMEnc}_{\mathrm{PK}}(m_b)$:

Let $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be an IND-CPA encryption scheme, $(\mathsf{GenSig}, \mathsf{Sign}, \mathsf{VerSig})$ be a strong one-time signature scheme.

$\mathsf{NMGen}(1^k)$:

1. Choose an error correcting code $C : \{0,1\}^k \to \{0,1\}^{20k}$ with the distance $6k$.
2. For $j \in [20k], b \in \{0,1\}$, run $\mathsf{Gen}(1^k)$ to generate key-pairs $(\text{PK}_j^b, \text{SK}_j^b)$.
3. Pick a random subset $S \subset [20k]$ of size k.
4. Set $\text{PK} = \Big(C, \{\text{PK}_j^b \mid j \in [20k], b \in \{0,1\}\}\Big)$ and $\text{SK} = \Big(S, \{\text{SK}_j^b \mid j \in [6k] \cup S, b \in \{0,1\}\}\Big)$.

$\mathsf{NMEnc}_{\text{PK}}(m)$ with $m \in \{0,1\}^n$:

1. Run $\mathsf{GenSig}(1^k)$ to generate $(\text{SKSIG}, \text{VKSIG})$. Parse VKSIG as an element $\text{VKSIG} \in \{0,1\}^k$. Let $C(\text{VKSIG}) := (v_1, \ldots, v_{20k})$, where $v_1, \ldots, v_{20k} \in \{0,1\}^{20k}$.
2. Parse m as $(m_1, \ldots, m_k)$ over $\text{GF}(2^\gamma)$, where $\gamma = \lceil n/k \rceil$. Choose a random polynomial p of degree $2k-1$ over $\text{GF}(2^\gamma)$ such that $p(20k+1) = m_1, \ldots, p(20k+k) = m_k$. Let $(s_1, \ldots, s_{20k}) = (p(1), \ldots, p(20k))$.
3. For $j \in [20k]$, compute the ciphertext $c_j \leftarrow \mathsf{Enc}_{\text{PK}_j^{v_j}}(s_j)$.
4. Compute $\sigma \leftarrow \mathsf{Sign}_{\text{SKSIG}}(\vec{c})$ where $\vec{c} = (c_1, \ldots, c_{20k})$, and output $[\vec{c}, \text{VKSIG}, \sigma]$.

$\mathsf{NMDec}_{\text{SK}}([\vec{c}, \text{VKSIG}, \sigma])$:

1. (`sig-check`) Verify the signature with $\mathsf{VerSig}_{\text{VKSIG}}[\vec{c}, \sigma]$.
2. (`decoding-check`) Let $\vec{c} = (c_j)$ and $(v_1, \ldots, v_{20k}) = C(\text{VKSIG})$. For $j \in [6k]$, compute $s_j = \mathsf{Dec}_{\text{SK}_j^{v_j}}(c_j)$. Apply the Berlekamp-Welch algorithm to $(s_1, \ldots, s_{6k})$ to recover a degree $(2k-1)$ polynomial p; if it fails, output $\perp$. Otherwise, let $(m_1, \ldots, m_k) = (p(20k+1), \ldots, p(20k+k))$ and $(w_1, \ldots, w_{20k})$ be $(p(1), \ldots, p(20k))$.
3. (`codeword-check`) Compute $s_j = \mathsf{Dec}_{sk_j^{v_j}}(c_j)$ for all $j \in S$. Check that $s_j = w_j$.
4. If all the checks accept, output the message m; otherwise output $\perp$.

Figure 3: NP-CPA PKE WITH MESSAGE LENGTH OF $\omega(k)$ USING RS-BASED RPE

Experiment $\mathsf{NME}_b^{(1)}$: $\mathsf{NME}_b^{(1)}$ proceeds exactly like NME_b, except we replace `sig-check` in NMDec with `sig-check`*:

> (`sig-check`*) Verify the signature with $\mathsf{VerSig}_{\text{VKSIG}}[\vec{c}, \sigma]$. Output $\perp$ if the signature fails to verify or if $\text{VKSIG} = \text{VKSIG}^*$.

Experiment $\mathsf{NME}_b^{(2)}$: $\mathsf{NME}_b^{(2)}$ proceeds exactly like $\mathsf{NME}_b^{(1)}$ except we replace NMDec with NMDec^*:

$\mathsf{NMDec}^*_{\text{SK}}([\vec{c}, \text{VKSIG}, \sigma])$:

1. (`sig-check`*) Verify the signature with $\mathsf{VerSig}_{\text{VKSIG}}[\vec{c}, \sigma]$. Output $\perp$ if the signature fails to verify or if $\text{VKSIG} = \text{VKSIG}^*$.
2. (`decoding-check`*) Let $\vec{c} = (c_j)$ and $C(\text{VKSIG}) = (v_1, \ldots, v_\ell)$. Let $X = (x_1, \ldots, x_{\lambda\ell})$ be the smallest distinct values such that $v_{x_i} \neq v^*_{x_i}$. Note there must be these values since C is an encoding with minimum distance λ. Compute $s_{x_i} = \mathsf{Dec}_{\text{SK}_{x_i}^{v_{x_i}}}(c_{x_i})$, $i = 1, \ldots, \lambda\ell$. Compute $w = (w_1, \ldots, w_\ell) \in \mathcal{W}$ such that $(w_{x_1}, \ldots, w_{x_{\lambda\ell}})$ is least $(1 - \frac{\delta'}{2})$-close to $(s_{x_1}, \ldots, s_{x_{\lambda\ell}})$ by running $\mathsf{D}'(X, (s_{x_1}, \ldots, s_{x_{\lambda\ell}}))$ based on the property of decoding from the partial view. If no such codeword exists, output $\perp$.

3. (`codeword-check*`) For all $j \in S$, check that $\mathsf{Dec}_{\mathrm{SK}_j^{v_j}}(c_j) = w_j$.

If all the checks accept, output the message m corresponding to the codeword w; else, output $\perp$.

Claim 1. *For $b \in \{0,1\}$, we have* $\left\{\mathsf{NME}_b(\Pi, A, k, p(k))\right\} \overset{c}{\approx} \left\{\mathsf{NME}_b^{(1)}(\Pi, A, k, p(k))\right\}$

Proof. This follows readily from the security of the signature scheme. □

Claim 2. *For $b \in \{0,1\}$, we have* $\left\{\mathsf{NME}_b^{(1)}(\Pi, A, k, p(k))\right\} \overset{s}{\approx} \left\{\mathsf{NME}_b^{(2)}(\Pi, A, k, p(k))\right\}$

Proof. We will show that both distributions are statistically close for all possible coin tosses in both experiments (specifically, those of NMGen, A and NMEnc) except for the choice of S in NMGen. Once we fix all the coin tosses apart from the choice of S, the output $(\psi_1, \ldots, \psi_{p(k)})$ of A_2 are completely determined and identical in both experiments $\mathsf{NME}_b^{(1)}$ and $\mathsf{NME}_b^{(2)}$.

Recall the guarantees we would like from NMDec and NMDec^*:

- On input a ciphertext that is an encryption of a message m under Π, both NMDec and NMDec^* will output m with probability 1.
- On input a ciphertext that is "close" to an encryption of a message m under Π, both NMDec and NMDec^* will output m with the same probability (the exact probability is immaterial) and $\perp$ otherwise.
- On input a ciphertext that is "far" from any encryption, then both NMDec and NMDec^* output $\perp$ with high probability.

To quantify and establish these guarantees, we consider the following promise problem (Π_Y, Π_N) that again refers to the underlying vector of plaintexts. An instance is a vector of ℓ entries each of which lies in $\{0,1\}^n \cup \perp$.

Π_Y (YES instances) — for some $w \in \mathcal{W}$, the instance equals w.

Π_N (NO instances) — either the first $\lambda\ell$ elements of the instance is $\delta'/2$-far from the first $\lambda\ell$ elements of every codeword in $\mathcal{W}$ or the entire instance is $\frac{\lambda\delta'}{2}$-far from every codeword in $\mathcal{W}$.

Valid encryptions correspond to the YES instances, while NO instances will correspond to "far" ciphertexts. To analyze the success probability of an adversary, we examine each ciphertext ψ it outputs with some underlying vector $\vec{M}$ of plaintexts (which may be a YES or a NO instance or neither) and show that both NMDec and NMDec^* agree on ψ with high probability. To facilitate the analysis, we consider two cases:

- If $\vec{M} \in \Pi_N$, then it fails the codeword checks in both decryption algorithms with high probability, in which case both decryption algorithms output $\perp$.

  Specifically, if the first $\lambda\ell$ elements of $\vec{M}$ is $\delta'/2$-far from the first $\lambda\ell$ elements of every codeword in $\mathcal{W}$ then the decoding check in NMDec rejects $\vec{M}$ with probability 1. Moreover, being $\delta'/2$-far from the first $\lambda\ell$ elements for every codeword implies that $\vec{M}$ have at least $(\delta'/2) \cdot \lambda\ell$ different positions, where c is some constant. Therefore, the codeword check in

NMDec^* rejects $\vec{M}$ with probability at least $1 - \left(1 - \frac{\delta'\lambda}{2}\right)^t$, since the condition implies that $\vec{M}$ is $\frac{\delta'\lambda}{2}$-far from every codeword. From Lemma 1, both δ' and λ are constant, and therefore with overwhelming probability in t, NMDec^* will reject $\vec{M}$ as well.

On the other hand, if $\vec{M}$ is $\lambda\delta'/2$-far from every codeword, both codeword checks in NMDec and NMDec^* rejects $\vec{M}$ with probability $1 - (1 - \frac{\delta'\lambda}{2})^t$

Therefore, both NMDec and NMDec^* reject $\vec{M}$ with with probability at least $1 - 2 \cdot (1 - \frac{\delta'\lambda}{2})^t$.

- If $\vec{M} \notin \Pi_N$, then both decryption algorithms always output the same answer for all choices of the set S, provided there is no forgery. Fix $\vec{M} \notin \Pi_N$ and a set S. Note that the decoding check in both NMDec and NMDec^* will be successful. This is because $\vec{M}$ is $(1-\lambda\delta'/2)$-close to w, and there are at most $(\lambda\delta'/2) \cdot \ell$ erroneous positions compared with some codeword in $\mathcal{W}$. This implies that any $\lambda\ell$ elements of $\vec{M}$ has at most $\frac{(\lambda\delta'/2)\cdot\ell}{\lambda\ell} = \frac{\delta'}{2}$ fraction error. Moreover, the codeword check is the same in both NMDec and NMDec^*. As such, both decryption algorithms output $\perp$ with exactly the same probability, and whenever they do not output $\perp$, they output the same message m.

From the above analysis, the two hybrids are statistically close. □

Claim 3. *For every ppt machine A, there exists a ppt machine B such that for $b \in \{0, 1\}$,*

$$\left\{\mathsf{NME}_b^{(2)}(\Pi, A, k, p(k))\right\} \equiv \left\{\mathsf{mIND}_b(E, B, k, \ell - t)\right\}$$

Proof. The machine B is constructed as follows: B participates in the experiment mIND_b (the "outside") while internally simulating $A = (A_1, A_2)$ in the experiment $\mathsf{NME}_b^{(2)}$.

- (pre-processing) Pick a random subset $S = \{u_1, \dots, u_t\}$ of $[\ell]$. Choose an ECC C, and run $\mathsf{GenSig}(1^k)$ to generate $(\text{SKSIG}^*, \text{VKSIG}^*)$ and set $(v_1^*, \dots, v_\ell^*) = C(\text{VKSIG}^*)$. Let $\phi : \{j \mid j \in [\ell] \setminus S\} \to [\ell - t]$ be a bijection.
- (key generation) B receives $\langle \text{PK}_1, \dots, \text{PK}_{\ell-t} \rangle$ from the outside and simulates NMGen as follows: for all $j \in [\ell], \beta \in [g]$,

$$(\text{PK}_j^\beta, \text{SK}_j^\beta) = \begin{cases} (\text{PK}_{\phi(j)}, \perp) & \text{if } \beta = v_j^* \text{ and } j \notin S \\ \mathsf{Gen}(1^k) & \text{otherwise} \end{cases}$$

- (message selection) Let (m^0, m^1) be the pair of messages A_1 returns. B chooses $(\alpha_1, \dots, \alpha_t) \leftarrow \Sigma^t$ uniformly at random and then computes

$$(w_1^0, \dots, w_\ell^0) \leftarrow \mathsf{R}(S, (\alpha_1, \dots, \alpha_t), m^0), \quad (w_1^1, \dots, w_\ell^1) \leftarrow \mathsf{R}(S, (\alpha_1, \dots, \alpha_t), m^1).$$

  Recall that R is the reconstruction algorithm of the underlying RPE scheme. For $j \in S$, let $\gamma_j = w_j^0 = w_j^1$.

  B forwards $(\langle m_1^0, \dots, m_{\ell-t}^0 \rangle, \langle m_1^1, \dots, m_{\ell-t}^1 \rangle)$ to the outside, where $m_{\phi(j)}^b = w_j^b$, for $j \in [\ell] \setminus S$.

- (ciphertext generation) B receives $\langle y_1, \ldots, y_{\ell-t}\rangle$ from the outside (according to the distribution $\mathsf{Enc}_{\mathrm{PK}_1}(m_1^b), \ldots, \mathsf{Enc}_{\mathrm{PK}_{\ell-t}}(m_{\ell-t}^b)$) and generates a ciphertext $[\vec{c}, \mathrm{VKSIG}^*, \sigma]$ as follows:

$$c_{i,j} = \begin{cases} y_{\phi(j)} & \text{if } j \notin S \\ \mathsf{Enc}_{\mathrm{PK}_j^{v_j^*}}(\gamma_j) & \text{otherwise} \end{cases}$$

  B then computes the signature $\sigma \leftarrow \mathsf{Sign}_{\mathrm{SKSIG}^*}(\vec{c})$ and forwards $[\vec{c}, \mathrm{VKSIG}^*, \sigma]$ to A_2. It is straight-forward to verify that $[\vec{c}, \mathrm{VKSIG}^*, \sigma]$ is indeed a random encryption of m_b under Π.

- (decryption) Upon receiving a sequence of ciphertexts $(\psi_1, \ldots, \psi_{p(k)})$ from A_2, B decrypts these ciphertexts using NMDec^* as in $\mathsf{NME}_b^{(2)}$. Note that to simulate NMDec^*, it suffices for B to possess the secret keys $\{\mathrm{SK}_j^\beta \mid \beta \neq v_j^* \text{ or } j \in S\}$, which B generated by itself. □

Combining the three claims, we conclude that for every ppt adversary A, there is a ppt adversary B such that for $b \in \{0,1\}$,

$$\begin{aligned}\left\{\mathsf{NME}_b(\Pi, A, k, p(k))\right\} &\overset{c}{\approx} \left\{\mathsf{NME}_b^{(1)}(\Pi, A, k, p(k))\right\} \\ &\overset{s}{\approx} \left\{\mathsf{NME}_b^{(2)}(\Pi, A, k, p(k))\right\} \equiv \left\{\mathsf{mIND}_b(E, B, k, \ell - t)\right\}\end{aligned}$$

Due to semantic security of the underlying encryption scheme, we have $\mathsf{mIND}_0(E, B, k, \ell - t) \overset{c}{\approx} \mathsf{mIND}_1(E, B, k, \ell - t)$, which concludes the proof.

7 Achieving Bounded-CCA2 Non-Malleability

We describe how our scheme may be modified to achieve non-malleability under a bounded-CCA2 attack. Recall that, informally, an encryption scheme is non-malleable against a q-bounded CCA2 attack if the adversary is allowed to query Dec adaptively at most $q(k)$ times in the non-malleable experiment. Our modification is the straight-forward analogue of the [CHH+07] modification of the [PSV06] scheme: We change the parameter $(n, \ell, \delta, t, \Sigma)$ of the underlying RPE scheme such that $t = a(k + q(k))$, where a is a constant such that $\left(1 - \frac{\lambda\delta'}{2}\right)^a \leq \frac{1}{2}$. See Figure 2 for more details.

We analyze security of the encryption scheme using the similar hybrid argument. We define the following hybrid experiments as before.

- Experiment $\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b^{(1)}$: This experiment proceeds exactly like $\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b$, except we replace `sig-check` in NMDec with `sig-check`* as described in Section 6.
- Experiment $\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b^{(2)}$: This experiment proceeds exactly like $\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b^{(1)}$ except we replace NMDec with NMDec^* as described in Section 6.

We note that $\left\{\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b(\Pi, A, k, p(k))\right\}$ and $\left\{\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b^{(1)}(\Pi, A, k, p(k))\right\}$ are computationally indistinguishable for each $b \in \{0,1\}$, which can be argued based on security of the signature scheme as in Claim 1. Moreover, $\left\{\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b^{(2)}(\Pi, A, k, p(k))\right\}$ and $\left\{\mathsf{mIND}_b(E, B, k, \ell - t)\right\}$ are identically distributed for each $b \in \{0,1\}$, which can be shown using the reduction in the proof of Claim 3. (Recall that the value $p(k)$ in the various $\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}$ experiments corresponds to the number of (mauled) ciphertexts that the adversary would come up with, after given the challenge ciphertext.) Therefore, we are only left to show the following claim to conclude the analysis.

Claim 4. *For $b \in \{0,1\}$, we have*

$$\left\{\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b^{(1)}(\Pi, A, k, p(k))\right\} \stackrel{s}{\approx} \left\{\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b^{(2)}(\Pi, A, k, p(k))\right\}$$

Proof. Let $q = q(k)$ and for a ciphertext c, let $\vec{M}_c$ denote the underlying plaintext matrix of c.

As before, we will show that both distributions are statistically close for all possible coin tosses in both experiments (specifically, those of NMGen, A and NMEnc) except for the choice of S in NMGen. Fix all the coin tosses apart from the choice of S. Here, however, unlike the case of chosen plaintext attacks, we cannot immediately deduce that the outputs of A_2 in both experiments are completely determined and identical, since they depend on the adaptively chosen queries to NMDec, and the answers depend on S. Still, the choice of S only affects whether the consistency checks accept or not; therefore, for each query, the number of possible responses of $\mathsf{NMDec}/\mathsf{NMDec}^*$ is at most two (since we fixed all the coin tosses except S). Moreover, if a query c is such that $\vec{M}_c \in \Pi_N$, NMDec and NMDec^* will both give only one response of $\perp$ with overwhelming probability, according to the analysis in Claim 2.

This leads us to consider a binary tree of depth q that corresponds informally to "unrolling" the q adaptive queries that A makes to $\mathsf{NMDec}/\mathsf{NMDec}^*$ in the experiments $\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b^{(1)}/\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b^{(2)}$. The root node of the tree corresponds to the first query A makes to $\mathsf{NMDec}/\mathsf{NMDec}^*$, and each edge from a node to its child is labeled with the answer of $\mathsf{NMDec}/\mathsf{NMDec}^*$ to the node's query. In particular, the tree is inductively built as follows:

- When A makes a query c with $\vec{M}_c \in \Pi_N$, we only consider the computation path corresponding to $\mathsf{NMDec}/\mathsf{NMDec}^*$ responding with $\perp$.
- When A makes a query c with $\vec{M}_c \notin \Pi_N$, we consider two computation paths, that is, one case of $\mathsf{NMDec}/\mathsf{NMDec}^*$ responding with a valid decryption (in which case the value returned is independent of S) and the other case of responding with $\perp$.
- The query at an internal node (except the root) corresponds to the query that A makes when following the computation path from the root to the node while $\mathsf{NMDec}/\mathsf{NMDec}^*$'s answers correspond to the labels of the edges in the path. Each leaf node contains $p(k)$ ciphertexts output by A at the end of the experiment.

Observe that the construction of the computation tree is completely deterministic and independent of the choice of S. Moreover, since NMDec and NMDec^* behave identically for queries c with $\vec{M}_c \notin \Pi_N$ as shown in Claim 2, the computation tree is $\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b^{(1)}$ is identical to that in $\mathsf{NME}\text{-}q\text{-}\mathsf{CCA}_b^{(2)}$. Note also that A makes at most q adaptive queries to NMDec, and therefore the total number of ciphertexts in the tree is at most $2^{q+1}p(k)$. The claim follows from combining the following two observations:

- Let $\mathsf{good}(S)$ be an event in which given the choice S, for every ciphertext c in the tree such that $\vec{M}_c \in \Pi_N$, both NMDec and NMDec^* output $\perp$. We have

  $$\Pr_S[\mathsf{good}(S)] \geq 1 - 2 \cdot (2^{q+1}p(k)) \cdot (1 - \lambda \cdot \delta'/2)^t \geq 1 - 2 \cdot (2^{q+1}p(k)) \cdot (1/2)^{k+q} = 1 - \mathsf{negl}(k).$$

  This follows from a union bound over these ciphertexts in the tree and the analysis in Claim 2.

- For every S such that $\mathsf{good}(S)$ is true, the outputs in both experiments are the same. This follows readily by induction on the queries made by A, and using the fact both NMDec and NMDec^* always output the same answer for any $\vec{M} \notin \Pi_N$ as explained in the analysis in Claim 2.

$\square$

References

[BDKM16] Marshall Ball, Dana Dachman-Soled, Mukul Kulkarni, and Tal Malkin. Non-malleable codes for bounded depth, bounded fan-in circuits. In *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part II*, pages 881–908, 2016.

[BGW88] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, pages 1–10, 1988.

[BN08] Mihir Bellare and Chanathip Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. *Journal of Cryptology*, 21(4):469–491, October 2008.

[BS99] Mihir Bellare and Amit Sahai. Non-malleable encryption: Equivalence between two notions, and an indistinguishability-based characterization. In Michael J. Wiener, editor, *Advances in Cryptology – CRYPTO'99*, volume 1666 of *Lecture Notes in Computer Science*, pages 519–536. Springer, August 1999.

[CDD+15] Ronald Cramer, Ivan Bjerre Damgård, Nico Döttling, Serge Fehr, and Gabriele Spini. Linear secret sharing schemes from error correcting codes and universal hash functions. Lecture Notes in Computer Science, pages 313–336. Springer, 2015.

[CDMW08] Seung Geol Choi, Dana Dachman-Soled, Tal Malkin, and Hoeteck Wee. Black-box construction of a non-malleable encryption scheme from any semantically secure one. In Ran Canetti, editor, *TCC 2008: 5th Theory of Cryptography Conference*, volume 4948 of *Lecture Notes in Computer Science*, pages 427–444. Springer, March 2008.

[CDMW16] Seung Geol Choi, Dana Dachman-Soled, Tal Malkin, and Hoeteck Wee. A black-box construction of non-malleable encryption from semantically secure encryption. 2016. Full version of [CDMW08], eprint/2016/720.

[CDTV16] Sandro Coretti, Yevgeniy Dodis, Björn Tackmann, and Daniele Venturi. Non-malleable encryption: Simpler, shorter, stronger. Lecture Notes in Computer Science, pages 306–335. Springer, 2016.

[CG14] Mahdi Cheraghchi and Venkatesan Guruswami. Non-malleable coding against bit-wise and split-state tampering. In Yehuda Lindell, editor, *TCC 2014: 11th Theory of Cryptography Conference*, volume 8349 of *Lecture Notes in Computer Science*, pages 440–464. Springer, February 2014.

[CHH$^+$07] Ronald Cramer, Goichiro Hanaoka, Dennis Hofheinz, Hideki Imai, Eike Kiltz, Rafael Pass, Abhi Shelat, and Vinod Vaikuntanathan. Bounded CCA2-secure encryption. In *Advances in Cryptology - ASIACRYPT 2007*, volume 4833 of *Lecture Notes in Computer Science*, 2007.

[CHK04] Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. In Christian Cachin and Jan Camenisch, editors, *Advances in Cryptology – EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 207–222. Springer, May 2004.

[CS03] Ronald Cramer and Victor Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *SIAM Journal on Computing*, 33(1):167–226, 2003.

[DDN00] Danny Dolev, Cynthia Dwork, and Moni Naor. Nonmalleable cryptography. *SIAM Journal on Computing*, 30(2):391–437, 2000.

[DGR99] Scott E. Decatur, Oded Goldreich, and Dana Ron. Computational sample complexity. *SIAM Journal on Computing*, 29(3):854–879, 1999.

[DMRY09] Dana Dachman-Soled, Tal Malkin, Mariana Raykova, and Moti Yung. Efficient robust private set intersection. In *Applied Cryptography and Network Security, 7th International Conference, ACNS 2009*, volume 5536 of *Lecture Notes in Computer Science*, 2009.

[FY92] Matthew K. Franklin and Moti Yung. Communication complexity of secure computation (extended abstract). In *24th Annual ACM Symposium on Theory of Computing*, pages 699–710. ACM Press, May 1992.

[GHKW16] Romain Gay, Dennis Hofheinz, Eike Kiltz, and Hoeteck Wee. Tightly cca-secure encryption without pairings. In *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I*, pages 1–27, 2016.

[GLOV12] Vipul Goyal, Chen-Kuei Lee, Rafail Ostrovsky, and Ivan Visconti. Constructing non-malleable commitments: A black-box approach. In *53rd Annual Symposium on Foundations of Computer Science*, pages 51–60. IEEE Computer Society Press, October 2012.

[GMM07] Yael Gertner, Tal Malkin, and Steven Myers. Towards a separation of semantic and CCA security for public key encryption. In *Proceedings of the 4th Theory of Cryptography Conference, TCC 2007*, volume 4392 of *Lecture Notes in Computer Science*, pages 434–455, 2007.

[GOSV14] Vipul Goyal, Rafail Ostrovsky, Alessandra Scafuro, and Ivan Visconti. Black-box non-black-box zero knowledge. In David B. Shmoys, editor, *46th Annual ACM Symposium on Theory of Computing*, pages 515–524. ACM Press, May / June 2014.

[HHK10] Javier Herranz, Dennis Hofheinz, and Eike Kiltz. Some (in)sufficient conditions for secure hybrid encryption. *Inf. Comput.*, 208(11):1243–1257, 2010.

[HJ12] Dennis Hofheinz and Tibor Jager. Tightly secure signatures and public-key encryption. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology – CRYPTO 2012*, volume 7417 of *Lecture Notes in Computer Science*, pages 590–607. Springer, August 2012.

[Kiy14] Susumu Kiyoshima. Round-efficient black-box construction of composable multi-party computation. In Juan A. Garay and Rosario Gennaro, editors, *Advances in Cryptology – CRYPTO 2014, Part II*, volume 8617 of *Lecture Notes in Computer Science*, pages 351–368. Springer, August 2014.

[KMO14] Susumu Kiyoshima, Yoshifumi Manabe, and Tatsuaki Okamoto. Constant-round black-box construction of composable multi-party computation protocol. In Yehuda Lindell, editor, *TCC 2014: 11th Theory of Cryptography Conference*, volume 8349 of *Lecture Notes in Computer Science*, pages 343–367. Springer, February 2014.

[KW15] Eike Kiltz and Hoeteck Wee. Quasi-adaptive NIZK for linear subspaces revisited. Lecture Notes in Computer Science, pages 101–128. Springer, 2015.

[Lam79] Leslie Lamport. Constructing digital signatures from a one-way function. Technical Report SRI-CSL-98, SRI International Computer Science Laboratory, 1979.

[LP12] Huijia Lin and Rafael Pass. Black-box constructions of composable protocols without set-up. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology – CRYPTO 2012*, volume 7417 of *Lecture Notes in Computer Science*, pages 461–478. Springer, August 2012.

[LPJY14] Benoît Libert, Thomas Peters, Marc Joye, and Moti Yung. Non-malleability from malleability: Simulation-sound quasi-adaptive NIZK proofs and CCA2-secure encryption from homomorphic signatures. In Phong Q. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology – EUROCRYPT 2014*, volume 8441 of *Lecture Notes in Computer Science*, pages 514–532. Springer, May 2014.

[LPJY15] Benoît Libert, Thomas Peters, Marc Joye, and Moti Yung. Compactly hiding linear spans - tightly secure constant-size simulation-sound QA-NIZK proofs and applications. Lecture Notes in Computer Science, pages 681–707. Springer, December 2015.

[Ms09] Steven Myers and abhi shelat. Bit encryption is complete. In *50th Annual Symposium on Foundations of Computer Science*, pages 607–616. IEEE Computer Society Press, October 2009.

[PSV06] Rafael Pass, Abhi Shelat, and Vinod Vaikuntanathan. Construction of a non-malleable encryption scheme from any semantically secure one. In *Advances in Cryptology - CRYPTO 2006*, volume 4117 of *Lecture Notes in Computer Science*, pages 271–289, 2006.

[PW08] Chris Peikert and Brent Waters. Lossy trapdoor functions and their applications. In Richard E. Ladner and Cynthia Dwork, editors, *40th Annual ACM Symposium on Theory of Computing*, pages 187–196. ACM Press, May 2008.

[PW09] Rafael Pass and Hoeteck Wee. Black-box constructions of two-party protocols from one-way functions. In *Proceedings of the 6th Theory of Cryptography Conference, TCC 2009*, volume 5444 of *Lecture Notes in Computer Science*, pages 403–418, 2009.

[Rom90] John Rompel. One-way functions are necessary and sufficient for secure signatures. In *Proceedings of the Twenty Second Annual ACM Symposium on Theory of Computing*, pages 387–394, 1990.

[RS10] Alon Rosen and Gil Segev. Chosen-ciphertext security via correlated products. *SIAM J. Comput.*, 39(7):3058–3088, 2010.

[Wee10] Hoeteck Wee. Black-box, round-efficient secure computation via non-malleability amplification. In *51st Annual Symposium on Foundations of Computer Science*, pages 531–540. IEEE Computer Society Press, October 2010.

[Wee12] Hoeteck Wee. Dual projective hashing and its applications - lossy trapdoor functions and more. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, volume 7237 of *Lecture Notes in Computer Science*, pages 246–262. Springer, April 2012.