Re-randomizable Encryption implies Selective Opening Security

Brett Hemenway* and Rafail Ostrovsky†

March 5, 2009

Abstract

In this paper, we present new general constructions of commitments and encryptions secure against a Selective Opening Adversary (SOA). Although it was recognized almost twenty years ago that SOA security was important, it was not until the recent breakthrough works of Hofheinz [Hof08] and Bellare, Hofheinz and Yilek [BHY09] that any progress was made on this fundamental problem.

The Selective Opening problem is as follows: suppose an adversary receives n commitments (or encryptions) of (possibly) correlated messages, and now the adversary can choose $n/2$ of the messages, and receive decommitments (or decryptions *and* the randomness used to encrypt them). Do the unopened commitments (encryptions) remain secure? A protocol which achieves this type of security is called *secure against a Selective Opening Adversary (SOA)*. This question arises naturally in the context of Byzantine Agreement and Secure Multiparty Computation, where an active adversary is able to eavesdrop on all the wires, and then choose a subset of players to corrupt. Unfortunately, the traditional definitions of security (IND-CPA,IND-CCA) do not guarantee security in this setting. In this paper:

- We formally define *re-randomizable* encryption and show that *every* re-randomizable encryption scheme gives rise to efficient encryptions secure against a selective opening adversary. (Very informally, an encryption is re-randomizable, if given any cyphertext, there is an efficient way to map it to an almost uniform re-encryption of the same underlying message).
- We formally define *re-randomizable* one-way functions and show that *every* re-randomizable one-way function family gives rise to efficient commitments secure against a Selective Opening Adversary.
- Applying our constructions to the known cryptosystems of El-Gamal, Paillier, and Goldwasser and Micali, we obtain IND-SO secure commitments and encryptions from the Decisional Diffie-Hellman (DDH), Decisional Composite Residuosity (DCR) and Quadratic Residuosity (QR) assumptions, that are either simpler or more efficient than existing constructions of Bellare Hofheinz and Yilek.
- Applying our general results to the Paillier Cryptosystem we demonstrate the first cryptosystem to achieve Semantic Selective Opening security from the DCR assumption.
- We give black-box constructions of Perfectly Binding SOA secure commitments, which is surprising given the negative results of Bellare, Hofheinz and Yilek.
- We define the notion of adaptive chosen ciphertext security (CCA-2) in the selective opening setting, and describe the first encryption scheme which is CCA-2 secure (and simultaneously SOA-secure).

Keywords: Public Key Encryption, Commitment, Selective Opening, Homomorphic Encryption, Chosen Ciphertext Security

*E-mail: bretth@math.ucla.edu

†E-mail: rafail@cs.ucla.edu

1 Introduction

In Byzantine agreement, and more generally in secure multiparty computation, it is often assumed that all parties are connected to each other via private channels. In practice, these private channels are implemented using a public-key cryptosystem. An active adversary in an MPC setting, however, has very different powers than an adversary in an IND-CPA or IND-CCA game. In particular, an active MPC adversary may view all the encryptions sent in a given round, and then choose to corrupt a certain fraction of the players, thus revealing the decryptions of those players' messages *and the randomness used to encrypt them*. The question is then whether the messages sent from the uncorrupted players remain secure. If the messages (and randomness) of all the players are chosen independently, then security in this setting follows immediately from the IND-CPA security of the underlying encryption. If, however, the messages are not chosen independently, the security does not immediately follow from the IND-CPA (or even IND-CCA) security of the underlying scheme. In fact, although this problem was first investigated over twenty years ago, it remains an open question whether IND-CPA (or IND-CCA) security implies this *Selective Opening* security.

A similar question may be asked regarded in terms of commitments as well. Suppose an adversary is allowed to see commitments to a number of related messages, the adversary may then choose a subset of the commitments for the challenger to decommit. Does this reveal any information about the unopened commitments? This question has applications to concurrent zero-knowledge proofs.

2 Previous Work

There have been many attempts to design encryption protocols that can be used to implement secure multiparty computation against an active adversary. The first protocols by Beaver and Haber [BH92] required interaction between the sender and receiver, required erasure and were fairly inefficient. The first non-interactive protocol was given by Canetti, Feige, Goldreich and Naor in [CFGN96]. In [CFGN96] the authors defined a new primitive called Non-Committing Encryption, and gave an example of such a scheme based on the RSA assumption. In [Bea97], Beaver extended the work of [CFGN96], and created adaptively secure key exchange under the Diffie-Hellman assumption. In subsequent work Damgård and Nielsen improved the efficiency of the schemes of Canetti et al. and Beaver, they were also able to obtain Non-Committing Encryption based on one-way trapdoor functions with invertible sampling. In [CHK05], Canetti, Halevi and Katz presented a Non-Committing encryption protocols with evolving keys.

In [CDNO97], Canetti, Dwork, Naor and Ostrovsky extended the notion of Non-Committing Encryption to a new protocol which they called Deniable Encryption. In Non-Committing Encryption schemes there is a simulator, which can generate non-committing ciphertexts, and later open them to any desired message, while in Deniable Encryption, valid encryptions generated by the sender and receiver can later be opened to any desired message. The power of this primitive made it relatively difficult to realize, and Canetti et al. were only able to obtain modest examples of Deniable Encryption and left it as an open question whether fully deniable schemes could be created.

The notions of security against an adaptive adversary can also be applied to commitments. In fact, according to [DNRS03] the necessity of adaptively-secure commitments was realized by 1985. Despite its utility, until recently there have been relatively few papers that directly address the question of commitments secure against a Selective Opening Adversary (SOA). The work of Dwork, Naor, Reingold and Stockmeyer [DNRS03] was the first to explicitly address the problem. In [DNRS03], Dwork et al. showed that non-interactive SOA-secure commitments can be used to create a 3-round zero-knowledge proof system for NP with negligible soundness error, and they gave constructions of a weak form of SOA-secure commitments, but leave open the question of whether general SOA-secure commitments exist.

The question of SOA-secure commitments was put on firm foundations by Hofheinz [Hof08] and Bellare, Hofheinz and Yilek in [BHY09]. In [BHY09], Bellare et al. distinguished between simulation-based and indistinguishability-based definitions of security, and gave a number of constructions and black-box separations. In particular, Hofheinz showed that in the simulation-based setting, non-interactive SOA-secure commitments cannot be realized in a black-box manner from standard cryptographic assumptions, but if interaction is allowed, they can be created from one-way permutations in a non-black-box manner. In the indistinguishability-based setting, they showed that any statistically-hiding scheme achieves this level of security, but that there is a black-box separation between perfectly-binding SOA-secure commitments and most standard cryptographic assumptions. In this paper, we build on the breakthrough results of [BHY09].

3 Our Contributions

In this paper we consider both encryptions and commitments secure against a selective opening adversary. In particular, we formalize the notion of **Re-Randomizable** Public-Key Encryption and we show that any re-randomizable encryption implies *both* commitments and encryptions secure against a selective opening adversary. We also define the notion of re-randomizable one-way functions and show that re-randomizable one-way functions imply commitments secure against a selective opening adversary. Using the definitions an terminology from [BHY09], we obtain black box constructions of IND-SO-COM commitments from any re-randomizable one-way function family, and IND-SO-ENC encryptions from any re-randomizable encryption. We also exhibit a black-box construction of perfectly-binding IND-SO-COM commitments from any re-randomizable one-way function family. This construction is somewhat surprising given the strong impossibility results for perfectly binding IND-SO-COM commitments in [BHY09]. Our constructions are simple, non-interactive and efficient.

On the encryption side, we show that re-randomizable encryption implies Lossy Encryption as defined in [PVW08], and expanded in [BHY09]. Combining this with the recent result of Bellare, Hofheinz and Yilek [BHY09] showing that Lossy Encryption is IND-SO-ENC secure, we have an efficient construction of IND-SO-ENC secure encryption from any re-randomizable encryption (which generalizes and extends previous results). Furthermore, both of these constructions retain the efficiency of the underlying re-randomizable encryption protocol.

Applying our results to the well-known cryptosystems of El-Gamal, Paillier, and Goldwasser-Micali, we obtain statistically-hiding or perfectly-binding IND-SO secure commitments and IND-SO encryptions based on the Decisional Diffie-Hellman problem, or the Decisional Composite Residuosity Problem, or the Quadratic Residuosity Problems respectively. We emphasize that in [BHY09] Bellare, Hofheinz and Yilek showed a Black-Box separation between perfectly-binding IND-SO-COM commitments and one-way permutations, one-way trapdoor permutations, and IND-CCA encryption, thus it is surprising to find Black-Box construction of perfectly-binding IND-SO secure commitments from re-randomizable functions.

While our primary contribution is the creation of IND-SO secure commitments and encryptions based on general assumption (of Re-Randomizable One-Way Functions and Re-Randomizable Public-Key Encryption respectively), our efficient construction of Lossy Encryption from any Re-Randomizable Encryption is interesting in its own right. In [PVW08] and [BHY09] it was shown that Lossy-Trapdoor Functions (as defined in [PW08]) imply Lossy Encryption, but it remained an open question whether in general the re-randomizable encryption implies Lossy Encryption. Here, we resolve this open question as well.

Applying our results to the Paillier Cryptosystem, we obtain the first cryptosystem which attains a strong, simulation-based form of semantic security under selective openings (SEM-SO-ENC security). This is the first construction of this type from the Decisional Composite Residuosity (DCR) assumption, and the most efficient known construction of SEM-SO-ENC secure encryption.

Finally, we present a definition of a chosen ciphertext (CCA-2) attack in the selective opening setting and create a the first public-key cryptosystem that satisfies this strengthened form of security. We note that our constructions are completely orthogonal to the recent work of Prabhakaran and Rosulek [PR07] creating RCCA Encryption. In their work, they create encryptions which satisfy a version of security against a chosen-ciphertext attack, while remaining re-randomizable. In this work, we use re-randomizable

(CPA secure) encryption to create Selective Opening secure encryption, and then use Selective Opening secure encryption (and other tools) to create a cryptosystem that retains its Selective Opening security against an adaptive chosen ciphertext attack.

4 Notation

If $f : X \to Y$ is a function, for any $Z \subset X$, we let $f(Z) = \{f(x) : x \in Z\}$.

If A is a PPT machine, then we use $a \leftarrow A$ to denote running the machine A and obtaining an output, where a is distributed according to the internal randomness of A. For a PPT machine A, we use $\mathsf{coins}(A)$ to denote the distribution of the internal randomness of A. So the distributions $\{a \leftarrow A\}$ and $\{r \leftarrow \mathsf{coins}(A) : a = A(r)\}$ are identical. If R is a set, we use $r \leftarrow R$ to denote sampling uniformly from R.

If X and Y are families of distributions indexed by a security parameter λ, we use $X \approx_s Y$ to mean the distributions X and Y are statistically close, i.e. for all polynomials p and sufficiently large λ we have

$$\sum_x |\Pr[X = x] - \Pr[Y = x]| < \frac{1}{p(\lambda)},$$

We use $X \approx_c Y$ to mean X and Y are computationally close, i.e. for all PPT adversaries A, for all polynomials p, then for all sufficiently large λ,

$$\left|\Pr[A^X = 1] - \Pr[A^Y = 1]\right| < \frac{1}{p(\lambda)}.$$

5 Re-randomizable Encryption

In many cryptosystems, given a ciphertext c, and a public-key it is possible to re-randomize the ciphertext to a new ciphertext c', such that c and c' are valid encryptions of the same plaintext, but they are statistically independent. Formally, we call a Public Key Cryptosystem given by algorithms (G, E, D) *re-randomizable* (RRPKC) if

- (G, E, D) is semantically-secure in the standard sense (IND-CPA).
- There is an efficient function ReRand such that if r' is chosen uniformly from $\mathsf{coins}(\mathsf{ReRand})$, and r_0, r_1 are chosen uniformly from $\mathsf{coins}(E)$, then the distributions

  $$\{r_0 \leftarrow \mathsf{coins}(E) : E(pk, m, r_0)\} \approx_s \{r' \leftarrow \mathsf{coins}(\mathsf{ReRand}) : \mathsf{ReRand}(E(pk, m, r_1), r')\}$$

  for all public keys pk and messages m, and randomness r_1.

We note that this definition of re-randomizable encryption provides a statistical re-randomization. It is possible to define re-randomizable encryption which satisfies perfect re-randomization (stronger) or computational re-randomization (weaker). Such definitions already exist in the literature (see for example [PR07],[Gro04],[JJS04]). Our constructions require statistical re-randomization, and do not go through under a computational re-randomization assumption.

There are many known examples of re-randomizable encryption. For example, if (G, E, D) is *homomorphic*, i.e. $E(pk, m_0, r_0) \cdot E(pk, m_1, r_1) = E(pk, m_0 + m_1, r^*)$, we can re-randomize by taking $\mathsf{ReRand}(pk, c, r') = c \cdot E(pk, 0, r')$. For all known homomorphic cryptosystems, (e.g. El-Gamal, Paillier, Damgård-Jurik, Goldwasser-Micali) we obtain re-randomizable encryption with this definition of ReRand.

We note, however, that this seems like a weaker requirement than homomorphic encryption, since re-randomization does not require any kind of group structure on the plaintext, or any method for combining ciphertexts.

6 Homomorphic Encryption

A Public Key Cryptosystem given by algorithms (G, E, D) is called *homomorphic* if

- The plaintext space forms a group X, with group operation $+$.
- The ciphertexts are members of a group Y.
- For all $x_0, x_1 \in X$, and for all $r_0, r_1 \in \mathsf{coins}(E)$, there exists an $r^* \in \mathsf{coins}(E)$ such that

$$E(pk, x_0 + x_1, r^*) = E(pk, x_0, r_0)E(pk, x_1, r_1).$$

Notice that we do not assume that the encryption is also homomorphic over the randomness, as is the case in most homomorphic encryption schemes, e.g. El-Gamal, Paillier, and Goldwasser-Micali. We also do not assume that the image $E(pk, X, R)$ is all of the group Y, only that $E(pk, X, R) \subset Y$. Since the homomorphic property implies closure, we have that $E(pk, X, R)$ is a semi-group. Notice also, that while it is common to use the word "homomorphic" to describe the cryptosystem, encryption is *not* a homomorphism in the mathematical sense.

We now show some basic properties from all homomorphic encryption schemes, these facts are commonly used, but since our definition is weaker than the (implicit) definitions of homomorphic encryption that appear in the literature, it is important to note that they hold under this definition as well.

- $E(pk, X, R)$ is a group.
- $E(pk, 0, R)$ is a subgroup of $E(pk, X, R)$.
- For all $x \in X$, $E(pk, x, R)$ is the coset $E(pk, x, r)E(pk, 0, R)$.
- For all $x_0, x_1 \in X$, $|E(pk, x_0, R)| = |E(pk, x_1, R)|$.
- If y is chosen uniformly from $E(pk, 0, R)$, then $yE(pk, x, r)$ is uniform in $E(pk, x, R)$.
- The group $E(pk, X, R) \simeq X \times E(pk, 0, R)$, and decryption is simply the homomorphism

$$E(pk, X, R) \to E(pk, X, R)/E(pk, 0, R) \simeq X.$$

We call a cryptosystem a *Homomorphic Public Key Cryptosystem* (HPKC), if the cryptosystem is IND-CPA secure, and homomorphic.

If we make the additional assumption that we can sample in a manner statistically close to uniform on the subgroup $E(pk, 0, R)$, then the cryptosystem (G, E, D) will be re-randomizable.

Definition 1. We call a Homomorphic Public Key Cryptosystem *Uniformly Sampleable* if there exists a PPT algorithm sample such that the output of $\mathsf{sample}(pk)$ is statistically close to uniform on the group $E(pk, 0, R)$.

We note, that for all known homomorphic cryptosystems we may define

$$\mathsf{sample}(pk) = \{r \leftarrow \mathsf{coins}(E) : E(pk, 0, r)\}.$$

It is not too hard to see that this property *does not* follow from the definition of Homomorphic Encryption.

7 Efficient Re-randomizable Encryption from Uniformly Sampleable Homomorphic Encryption

The scheme described above only allows commitment to single bits. If the underlying cryptosystem (G, E, D), can encrypt more than one bit at a time, we can increase the efficiency of this system, by simply putting $c_0, c_1, \ldots, c_n$ into the public key, and a commitment to i will be $\mathsf{ReRand}(pk, c_i, r)$. In most cases, however, we can increase the size of the committed messages without increasing the public-key.

In particular, if $(G, E, D, \mathsf{sample})$ is a Uniformly Sampleable Homomorphic Encryption scheme and $\mathbb{Z}/N\mathbb{Z} \hookrightarrow X$. Then, we can commit to elements in $\{0, 1, \ldots, N-1\}$ instead of $\{0, 1\}$ by simply taking

Parameter Generation:

- $(pk, sk) \leftarrow G(1^\lambda)$,
- $r \leftarrow \mathsf{coins}(E)$,
- $c = E(pk, b, r)$,

The public parameters are (pk, c).

Encryption:

- $r' \leftarrow \mathsf{coins}(\mathsf{sample})$.
- $c' \leftarrow \mathsf{sample}(pk, r')$.
- return $c^a \cdot c'$.

Decryption:
To decrypt a ciphertext c, simply return $D(c)$.

Now, if $c = E(pk, 0, r)$ the scheme is lossy, since all encryptions will be uniformly distributed in the subgroup $E(pk, 0, R)$, while if $c = E(pk, 1, r)$ the scheme is injective by the correctness of the decryption algorithm. This is the natural construction when working with the Paillier or Damgård-Jurik cryptosystems. We must use caution when applying this to El-Gamal, since the inverse map $\mathbb{Z}/N\mathbb{Z} \hookrightarrow X$ is not efficiently computable (it's the discrete log). In this context, it will not be a problem, since we never have to compute the inverse to decommit. If, on the other hand, we wanted to view this as an encryption scheme instead of a commitment scheme, then this lack of inverse would be an issue. Fortunately, there is a well known scheme to create a re-randomizable encryption from the DDH assumption that is only a slight modification of the original El-Gamal scheme. See [NP01], [PVW08] or [BHY09] for a description of this scheme. We stress, however, that "plain" El-Gamal is re-randomizable, however, it is slightly less efficient than this modification.

8 Re-randomizable One-Way Functions

A family of functions $\mathcal{F}$, indexed by a security parameter λ is called a *re-randomizable one-way function* family if the following conditions are satisfied

- **Efficiently Computable:** For all $f \in \mathcal{F}$, the function

  $$f : M \times R \to Y$$

  is efficiently computable.

- **One-Way:** For all PPT adversaries $A = (A_1, A_2)$,

  $$\Pr\left[f \leftarrow \mathcal{F}; (m_0, m_1, st) \leftarrow A_1(f); b \leftarrow \{0,1\}; r \leftarrow R; b' \leftarrow A_2(f(m_b, r), st) : b = b'\right] < \frac{1}{2} + \nu$$

  for some negligible function ν (of λ).

- **Injective on the first input:** For all $m \neq m' \in M$, and $r, r' \in R$,

$$f(m, r) \neq f(m', r').$$

This is equivalent to the statement

$$f(m, R) \cap f(m', R) = \emptyset$$

for all $m \neq m'$.

- **Re-randomizable:** For each f, there exists and efficient function ReRand such that for all $m \in M$ and $r_0 \in R$, we have

$$\{r \leftarrow R; f(m, r)\} \approx_s \{r \leftarrow \mathsf{coins}(\mathsf{ReRand}); \mathsf{ReRand}(f(m, r_0), r)\}.$$

It is easy to see that the encryption algorithm from a re-randomizable encryption scheme is immediately a re-randomizable one-way function. We note, however, that re-randomizable one-way functions are a significantly weaker primitive since we do not require any kind of trapdoor.

9 Commitments from Re-randomizable One-Way Functions

We begin by describing a construction of a simple bit commitment scheme that arises from any re-randomizable one-way function. Let $\mathcal{F}$ be a re-randomizable one-way function family. Then we define

Parameter Generation:

- $(f, \mathsf{ReRand}) \leftarrow \mathcal{F}(1^\lambda)$,
- $r_0, r_1 \leftarrow R$,
- $c_0 = f(b_0, r_0)$,
  $c_1 = f(b_1, r_1)$.

The public parameters are $(f, \mathsf{ReRand}, c_0, c_1)$.

Commitment:

- $r' \leftarrow \mathsf{coins}(\mathsf{ReRand})$,
- $\mathrm{Com}(b, r') = \mathsf{ReRand}(c_b, r')$.

Decommitment:
To decommit, simply reveal the randomness r'.

This scheme has a number of nice properties. If $b_0 = b_1$ then the scheme is statistically hiding by the properties of ReRand. Alternatively, if $b_0 \neq b_1$ then the scheme is perfectly binding by the injectivity of f on its first input. Now, the two modes are indistinguishable by the one-wayness of the f, combining this with the preceding observations, we obtain the following consequences. If $b_0 = b_1$ then the scheme is computationally binding, and if $b_0 \neq b_1$ the scheme is computationally hiding.

The security analysis is very straightforward, but as this will be the foundation of all our constructions we include it.

Lemma 1. If $b_0 = b_1$, the scheme described in Section 9 is statistically hiding and if $b_0 \neq b_1$, this scheme is perfectly binding.

Proof. If $b_0 = b_1$, the distributions

$$\{r' \leftarrow \mathsf{coins}(\mathrm{Com}) : \mathrm{Com}(0, r')\} \approx_s \{s' \leftarrow \mathsf{coins}(\mathrm{Com}) : Com(1, s')\},$$

by the definition of ReRand. On the other hand, if $b_0 \neq b_1$, $\mathrm{Com}(0, r) \in f(b_0, R)$, and $\mathrm{Com}(1, s) \in f(b_1, R)$, but by the injectivity on the first input, these sets are disjoint . $\square$

Lemma 2. The schemes when $b_0 = b_1$ and when $b_0 \neq b_1$ are computationally indistinguishable.

Proof. This is exactly the one-way property of f. $\square$

Corollary 1. If $b_0 = b_1$, this scheme is computationally binding, and if $b_0 \neq b_1$, this scheme is computationally hiding.

Proof. Since the scheme is perfectly binding when $b_0 \neq b_1$, breaking the binding property amounts to a proof that $b_0 = b_1$. Since the two modes are computationally indistinguishable, no computationally bounded adversary can create such a "proof." Similarly, since the scheme is perfectly hiding when $b_0 = b_1$, breaking the hiding property amounts to showing that $b_0 \neq b_1$, since the two modes are computationally indistinguishable, no probabilistic polynomial-time adversary can break the hiding property. $\square$

The ability to choose whether the commitment scheme is statistically hiding or perfectly binding is a valuable property, but it is the fact that this choice can be hidden *from the committer* that makes this construction truly useful.

10 Selective Opening Secure Commitments

10.1 Definitions

Definition 2. (Indistinguishability under selective openings/IND-SO-COM).

Let Com be a commitment scheme, we say that Com is indistinguishable under selective openings (IND-SO-COM secure) if for every PPT message distribution M and every PPT adversary A, we have that

$$\left|\Pr\left[A^{\textsf{ind-so-real}} = 1\right] - \Pr\left[A^{\textsf{ind-so-ideal}} = 1\right]\right| < \nu$$

for some negligible function ν, and where the games ind-so-real and ind-so-ideal are defined as follows

IND-SO-COM (Real)	IND-SO-COM (Ideal)
• $(m_1, \ldots, m_n) \leftarrow M$	• $(m_1, \ldots, m_n) \leftarrow M$
• $r_1, \ldots, r_n \leftarrow \mathsf{coins}(\mathrm{Com})$	• $r_1, \ldots, r_n \leftarrow \mathsf{coins}(\mathrm{Com})$
• $I \leftarrow A((\mathrm{Com}(m_1, r_1), \ldots, \mathrm{Com}(m_n, r_n))$	• $I \leftarrow A((\mathrm{Com}(m_1, r_1), \ldots, \mathrm{Com}(m_n, r_n))$
• $b \leftarrow A(\mathrm{Dec}(\mathrm{Com}(m_i, r_1))_{i \in I}, (m_1, \ldots, m_n))$	• $(m'_1, \ldots, m'_n) \leftarrow M \vert M_I$
	• $b \leftarrow A(\mathrm{Dec}(\mathrm{Com}(m_i, r_i))_{i \in I}, (m'_1, \ldots, m'_n))$

More explicitly, in the real game,

- The challenger samples messages $(m_1, \ldots, m_n) \leftarrow M$, from the joint message distribution.
- The challenger generates randomness $r_1, \ldots, r_n \leftarrow \mathsf{coins}(\mathrm{Com})$.
- The challenger sends $(\mathrm{Com}(m_1, r_1), \ldots, \mathrm{Com}(m_n, r_n))$ to A.
- The adversary A responds with a subset $I \subset \{1, \ldots, n\}$, with $|I| = n/2$.
- The challenger decommits $(\mathrm{Com}(m_i, r_i))_{i \in I}$.
- The challenger sends $(m_1, \ldots, m_n)$ to the adversary.

- The adversary outputs a bit b.

In the ideal game,

- The challenger samples messages $(m_1, \dots, m_n) \leftarrow M$, from the joint message distribution.
- The challenger generates randomness $r_1, \dots, r_n \leftarrow \mathsf{coins}(\mathrm{Com})$.
- The challenger sends $(\mathrm{Com}(m_1, r_1), \dots, \mathrm{Com}(m_n, r_n))$ to A.
- The adversary A responds with a subset $I \subset \{1, \dots, n\}$, with $|I| = n/2$.
- The challenger decommits $(\mathrm{Com}(m_i, r_i))_{i \in I}$.
- The challenger samples a new vector $m' \leftarrow M|M_I$, from M conditioned on the fact that $m_i = m_i'$ for $i \in I$, and sends M' to A.
- The adversary outputs a bit b.

10.2 IND-SO-COM Constructions from Re-randomizable Encryption

To construct an IND-SO-COM secure commitment scheme, it is enough to create a statistically hiding commitment scheme, since Bellare, Hofheinz and Yilek showed

Theorem 1. (Theorem 6 From [BHY09]).
Statistically-hiding commitment schemes are IND-SO-COM secure.

Proof. We follow the general form of the proof from [BHY09], but by restricting ourselves to non-interactive commitments we can slightly simplify the exposition. We begin by defining an (inefficient) algorithm called opener, which tries to open a commitment c to a specified message m. In particular

$$\mathsf{opener}(c, m) = \begin{cases} r & \text{s.t. } \mathrm{Com}(m, r) = c, \\ \perp & \text{if no such } r \text{ exists .} \end{cases}$$

Now, we proceed in a sequence of games. Let Game_{-1} be the real IND-SO-COM game. Let Game_0 be the game, where the challenger uses opener to decommit the commitments $(\mathrm{Com}(m_i, r_i))_{i \in I}$. Notice that the views of the adversary in Game_{-1} and Game_0 are identical (but Game_0 is no longer efficiently implementable). In particular

$$\Pr[A^{\mathrm{Game}_{-1}} = 1] = \Pr[A^{\mathrm{Game}_0} = 1].$$

Next we describe Game j for $j \in [n]$. The only difference between Game_j and Game_0 is that in Game_j for $i \leq j$, the challenger sends the vector

$$(\mathrm{Com}(\delta, r_1), \dots, \mathrm{Com}(\delta, r_{j-1}), \mathrm{Com}(m_j, r_j), \dots, \mathrm{Com}(m_n, r_n)),$$

for some fixed dummy message δ. For concreteness, we may set $\delta = 0^\lambda$. Now, the only difference between Game_j and Game_{j-1} is whether the jth commitment is a commitment to m_j or δ. Since Com is *statistically*-hiding, even an *unbounded* adversary has only a negligible probability of distinguishing the two cases. Thus by the triangle inequality

$$|\Pr[A^{\mathrm{Game}_0} = 1] - \Pr[A^{\mathrm{Game}_n} = 1]| < n \cdot \nu = \text{negligible},$$

where ν is the probability that an *unbounded* adversary breaks the hiding property of Com. Thus we obtain

$$|\Pr[A_{real}^{IND-SO-COM} = 1] - \Pr[A^{\mathrm{Game}_n} = 1] = \text{negligible}.$$

Finally, we notice that in Game_n, all the commitments are independent of the message $(m_1, \ldots, m_n)$, so we can repeat the above argument, starting with the ideal IND-SO-COM game, instead of the real game. Thus we obtain

$$|\Pr[A_{ideal}^{IND-SO-COM} = 1] - \Pr[A^{\text{Game}_n} = 1] = \text{negligible}.$$

Thus

$$|\Pr[A_{ideal}^{IND-SO-COM} = 1] - \Pr[A_{real}^{IND-SO-COM} = 1]| = \text{negligible}.$$

□

The commitment scheme constructed in §9 is statistically hiding when $b_0 = b_1$, so we obtain the following corollary

Corollary 2. Re-randomizable one-way functions imply non-interactive IND-SO-COM secure commitments.

Since Re-randomizable encryptions imply re-randomizable one-way functions, we have

Corollary 3. Re-randomizable encryption implies non-interactive IND-SO-COM secure commitments.

Perhaps more interesting is the case when $b_0 \neq b_1$. The commitment scheme constructed in §9 is no longer perfectly hiding, so Theorem 1 doesn't apply. In this case, we can still achieve IND-SO-COM security, by using the indistinguishability of the two modes. Roughly, this follows because an IND-SO-COM adversary must have similar probabilities of success against both modes, otherwise it could be used to distinguish the modes. Thus we arrive at the following Corollary.

Corollary 4. Re-randomizable one-way functions imply perfectly-binding IND-SO-COM secure commitments.

Since Re-randomizable encryptions imply re-randomizable one-way functions, we have

Corollary 5. Re-randomizable encryption implies perfectly binding non-interactive IND-SO-COM secure commitments.

Proof. We proceed via contradiction. Suppose there exists an IND-SO-COM adversary A that succeeds against the protocol with probability $\frac{1}{2} + \epsilon$ when $b_0 = b_1$. We will use A to construct a distinguisher D for the one-way game against the underlying re-randomizable one-way function f. In the one-wayness game against f, the challenger samples a function f and sends it to D. D will respond by sending $\{0, 1\}$ to the one-wayness challenger, and the one-wayness challenger samples $r \leftarrow R$ and sends $e = f(b, r)$ to D. Now, D samples $r' \leftarrow R$, and generates $e' = f(0, r')$. Now, D creates an instantiation of the commitment protocol setting $c_0 = e, c_1 = e'$, and plays the IND-SO-COM game with the adversary A. If A wins, D guesses $b = 1$, and if A loses, D guesses $b = 0$. From Theorem 1 we know that if $b = 0$ then A succeeds with probability ν for some negligible function ν. On the other hand, by hypothesis, if $b = 1$, then A wins the IND-SO-COM game with probability ϵ. Now

$$\begin{aligned}
\Pr[D \text{ wins }] &= \Pr[b = 1 \cap \ A \text{ wins }] + \Pr[b = 0 \cap \ A \text{ loses }] \\
&= \Pr[A \text{ wins}|b = 1] \Pr[b = 1] + \Pr[A \text{ loses}|b = 0] \Pr[b = 0] \\
&= \frac{1}{2}\left(\frac{1}{2} + \epsilon + \frac{1}{2} - \nu\right) \\
&= \frac{1}{2} + \frac{\epsilon - \nu}{2}.
\end{aligned}$$

Since ϵ is non-negligible, and ν is negligible, D breaks the one-way property of f. □

This result is perhaps surprising, since [BHY09] showed a black-box separation between most known cryptographic primitives and Perfectly Binding IND-SO-COM secure commitments.

11 Selective Opening Secure Encryption

11.1 Preliminaries

The notions of security against a Selective Decryption Adversary apply naturally to the encryption setting as well (this was originally formalized in [BHY09]). The security real and ideal security games are defined as in the case of commitments (see Section 10.1), except that in the opening phase, the challenger responds with the decryptions and the randomness used instead of the decommitments and randomness.

Definition 3. (Indistinguishability under selective openings/IND-SO-ENC).
Let (G, E, D) be a Public Key Cryptosystem (PKC), we say that (G, E, D) is indistinguishable under selective openings (IND-SO-ENC secure) if for every PPT message distribution M and every PPT adversary A, we have that

$$\left|\Pr\left[A^{\textsf{ind-so-real}} = 1\right] - \Pr\left[A^{\textsf{ind-so-ideal}} = 1\right]\right| < \nu$$

for some negligible function ν, and where the games $\textsf{ind-so-real}$ and $\textsf{ind-so-ideal}$ are defined as follows

IND-SO-ENC (Real)	IND-SO-ENC (Ideal)
• $(m_1, \ldots, m_n) \leftarrow M$	• $(m_1, \ldots, m_n) \leftarrow M$
• $r_1, \ldots, r_n \leftarrow \textsf{coins}(E)$	• $r_1, \ldots, r_n \leftarrow \textsf{coins}(E)$
• $I \leftarrow A((E(m_1, r_i), \ldots, E(m_n, r_n))$	• $I \leftarrow A((E(m_1, r_i), \ldots, E(m_n, r_n))$
• $b \leftarrow A(((m_i, r_i))_{i \in I}, (m_1, \ldots, m_n))$	• $(m'_1, \ldots, m'_n) \leftarrow M \vert M_I$
	• $b \leftarrow A((m_i, r_i))_{i \in I}, (m'_1, \ldots, m'_n))$

More explicitly, in the real game,

- The challenger samples messages $(m_1, \ldots, m_n) \leftarrow M$, from the joint message distribution.
- The challenger generates randomness $r_1, \ldots, r_n \leftarrow \textsf{coins}(E)$.
- The challenger sends $(E(m_1, r_1), \ldots, E(m_n, r_n)$ to A.
- The adversary A responds with a subset $I \subset \{1, \ldots, n\}$, with $|I| = n/2$.
- The challenger reveals *both* m_i and r_i for $i \in I$.
- The challenger sends $(m_1, \ldots, m_n)$ to the adversary.
- The adversary outputs a bit b.

In the ideal game,

- The challenger samples messages $(m_1, \ldots, m_n) \leftarrow M$, from the joint message distribution.
- The challenger generates randomness $r_1, \ldots, r_n \leftarrow \textsf{coins}(E)$.
- The challenger sends $(E(m_1, r_1), \ldots, E(m_n, r_n))$ to A.
- The adversary A responds with a subset $I \subset \{1, \ldots, n\}$, with $|I| = n/2$.
- The challenger reveals *both* m_i and r_i for $i \in I$.

- The challenger samples a new vector $m' \leftarrow M|M_I$, from M conditioned on the fact that $m_i = m'_i$ for $i \in I$, and sends M' to A.

- The adversary outputs a bit b.

We emphasize that the challenger reveals both the messages m_i and the randomness r_i for the selected messages. If the challenger only revealed the messages m_i, this type of security would follow immediately from IND-CPA security.

12 Lossy Encryption

In [PVW08], Peikert, Vaikuntanathan and Waters defined Dual-Mode Encryption, a type of cryptosystem with two types public-keys, injective keys on which the cryptosystem behaves normally and "lossy" or "messy" keys on which the system loses information about the plaintext. In particular they require that the encryptions of any two plaintexts under a lossy key yield distributions that are statistically close, yet injective and lossy keys remain computationally indistinguishable.

In [BHY09] Bellare, Hofheinz and Yilek define *Lossy Encryption*, expanding on the definitions of Dual-Mode Encryption in [PVW08], and Meaningful/Meaningless Encryption in [KN08]. At a high level, a 'lossy' (or 'messy' in the terminology of [PVW08]) cryptosystem is one which has two types of public keys which specify two different modes of operation. In the normal mode, encryption is injective, while in the lossy (or 'messy') mode, the ciphertexts generated by the encryption algorithm are independent of the plaintext. We also require that no efficient adversary can distinguish normal keys from lossy keys. In [BHY09], they also require openability, which basically allows the decryptor to decrypt a ciphertext generated from a lossy key to *any* plaintext.

Definition 4. Formally, an *lossy public-key encryption scheme* is a tuple $(G_{\mathsf{inj}}, G_{\mathsf{lossy}}, E, D)$ of polynomial-time algorithms such that

- $G_{\mathsf{inj}}(1^\lambda)$ outputs keys (pk, sk), keys generated by G_{inj} are called *injective keys.*

- $G_{\mathsf{lossy}}(1^\lambda)$ outputs keys $(pk_{\mathsf{lossy}}, sk_{\mathsf{lossy}})$, keys generated by G_{lossy} are called *lossy keys.*

Additionally, the algorithms must satisfy the following properties:

1. *Correctness on injective keys.* For all $x \in X$,

$$\Pr\left[(pk, sk) \leftarrow G_{\mathsf{inj}}(1^\lambda); r \leftarrow \mathsf{coins}(E) : D(sk, E(pk, x, r)) = x\right] = 1.$$

2. *Indistinguishability of keys.* This basically says that the the pk in lossy mode and injective mode are computationally indistinguishable. Specifically, if $\mathrm{proj} : (pk, sk) \mapsto pk$ is the projection map, then the two distributions

$$\{\mathrm{proj}(G_{\mathsf{inj}}(1^\lambda))\} \approx_c \{\mathrm{proj}(G_{\mathsf{lossy}}(1^\lambda))\}$$

3. *Lossiness of lossy keys.* If $(pk_{\mathsf{lossy}}, sk_{\mathsf{lossy}}) \leftarrow G_{\mathsf{lossy}}$, then for all $x_0, x_1 \in X$, the two distributions $E(pk_{\mathsf{lossy}}, x_0, R)$ and $E(pk_{\mathsf{lossy}}, x_1, R)$ are statistically close, i.e. the statistical distance is negligible in λ.

4. *Openability.* If $(pk_{\mathsf{lossy}}, sk_{\mathsf{lossy}}) \leftarrow G_{\mathsf{lossy}}$, and $r \leftarrow \mathsf{coins}(E)$, then for all $x_0, x_1 \in X$ with all but negligible probability, there exists an $r' \in \mathsf{coins}(E)$, such that $E(pk_{\mathsf{lossy}}, x_0, r) = E(pk_{\mathsf{lossy}}, x_1, r')$. While this is a statistical property that follows immediately from property (3), it is convenient, to state it explicitly, and to rephrase it in terms of an algorithm. We require that with all but negligible probability there is an (unbounded) algorithm opener that can open a lossy ciphertext to *any* plaintext.

Although the Openability property is implied by property (3), it is useful to include it explicitly because it simplifies the exposition somewhat. It also generalizes nicely, and in [BHY09] they show that if the algorithm opener is efficient, then the encryption scheme is actually SEM-SO-ENC secure (instead of only IND-SO-ENC).

We do not explicitly assume, that the scheme is IND-CPA secure, and in fact, the semantic security of the scheme follows from the indistinguishability of keys and the lossiness of the lossy keys, since for any $x_0, x_1 \in X$,

$$E(\pi(G_{\mathsf{inj}}(1^\lambda)), x_0, R) \approx_c E(\pi(G_{\mathsf{lossy}}(1^\lambda)), x_0, R)) \approx_s E(\pi(G_{\mathsf{lossy}}(1^\lambda)), x_1, R) \approx_c E(\pi(G_{\mathsf{inj}}(1^\lambda)), x_1, R).$$

In [BHY09] it was shown that Lossy Encryption can be constructed in a straightforward manner from Lossy-Trapdoor Functions, in fact, they simply observe that the CPA-secure system given in [PW08] is a Lossy Encryption.

Next, they showed

Theorem 2. Lossy Encryption is IND-SO-ENC secure.

Proof. This is proven in [BHY09].
The proof follows the same structure as the proof of Theorem 1. □

Thus to create IND-SO-ENC secure encryptions, it suffices to construct Lossy Encryption.

12.1 Re-randomizable Encryption Implies Lossy Encryption

To create Lossy Encryption from re-randomizable encryption, we parallel the construction of commitments in §9. As before, we let (G, E, D) be a re-randomizable public-key cryptosystem, and we create Lossy Encryption $(\bar{G}_{\mathsf{inj}}, \bar{G}_{\mathsf{lossy}}, \bar{E}, \bar{D})$ as follows:

- $\bar{G}_{\mathsf{inj}}(1^\lambda)$ runs $\bar{G}(1^\lambda)$, generating a pair (pk, sk). Then G_{inj} picks $r_0, r_1 \leftarrow \mathsf{coins}(E)$, and generates $e_0 = E(pk, 0, r_0)$, $e_1 = E(pk, 1, r_1)$. $\bar{G}_{\mathsf{inj}}$ returns $(\bar{pk}, \bar{sk}) = ((pk, e_0, e_1), sk)$.
- $\bar{G}_{\mathsf{lossy}}(1^\lambda)$ runs $\bar{G}(1^\lambda)$, generating a pair (pk, sk). Then G_{lossy} picks $r_0, r_1 \leftarrow \mathsf{coins}(E)$, and generates $e_0 = E(pk, 0, r_0)$, $e_1 = E(pk, 0, r_1)$. $\bar{G}_{\mathsf{lossy}}$ returns $(\bar{pk}, \bar{sk}) = ((pk, e_0, e_1), sk)$.
- $\bar{E}(\bar{pk}, b, r') = \mathsf{ReRand}(pk, e_b, r')$ for $b \in \{0, 1\}$.
- $\bar{D}(\bar{sk}, c)$, simply outputs $D(sk, c)$.

To see that this is a lossy encryption we notice that under an injective key it is clearly injective by the correctness of the decryption algorithm D, while in lossy mode, it is statistically lossy by the properties of the ReRand function. The proof that this Lossy Encryption is straightforward and we check the details here.

1. *Correctness on injective keys.* This follows immediately from the correctness of E.
2. *Indistinguishability of keys.* This follows immediately from the IND-CPA security of (G, E, D).
3. *Lossiness of lossy keys.* Notice that under a lossy public-key $\bar{pk}$, e_0 and e_1 are both encryptions of zero, so $\bar{E}(\bar{pk}, b, r)$ will also be an encryption of zero for $b \in \{0, 1\}$. By the properties of ReRand, we have that the distributions $\{\bar{E}(\bar{pk}, 0, r)\}$ and $\{\bar{E}(\bar{pk}, 1, r)\}$ are statistically close, which is exactly what is required for a key to be "lossy".
4. *Openability.* Under a lossy public-key, $\bar{E}(\bar{pk}, b, r') = \mathsf{ReRand}(E(pk, 0, r_b), r')$. Since r' is chosen uniformly from $\mathsf{coins}(\mathsf{ReRand})$, the properties of ReRand guarantee that the distributions $\mathsf{ReRand}(E(pk, 0, r_b), r')$ and $\mathsf{ReRand}(E(pk, 0, r_{1-b}, r''))$ are statistically close. That there exists an r'' such that $\mathsf{ReRand}(E(pk, 0, r_b), r') = \mathsf{ReRand}(E(pk, 0, r_{1-b}), r'')$ then follows from lemma 3.

Lemma 3. If R is a random variable, and $f : R \to X$, $g : R \to Y$ and

$$\sum_{z \in X \cup Y} \Pr\left[r \leftarrow R : f(r) = z\right] - \Pr\left[r \leftarrow R : g(r) = z\right] = \nu,$$

then

$$\Pr\left[r \leftarrow R : \nexists r' \in R \text{ such that } f(r) = g(r')\right] < \nu.$$

Proof. It suffices to notice that

$$\begin{aligned}
\nu &= \sum_{z \in X \cup Y} \Pr\left[r \leftarrow R : f(r) = z\right] - \Pr\left[r \leftarrow R : g(r) = z\right] \\
&\geq \sum_{z \in X \setminus Y} \Pr\left[r \leftarrow R : f(r) = z\right] - \Pr\left[r \leftarrow R : g(r) = z\right] \\
&= \Pr\left[r \leftarrow R : \nexists r' \in R \text{ such that } f(r) = g(r')\right]
\end{aligned}$$

□

13 A Simulation-Based Definition of Security

While we have focused on an indistinguishability-based definition of security for commitments and encryptions, it is also possible to give a simulation-based definition. Roughly, this says that anything an adversary can learn by playing the Selective Opening game with the challenger can be efficiently simulated by a simulator that sees only I and $(m_i)_{i \in I}$, and never sees the ciphertexts at all. This is called SEM-SO-ENC security. While it appears that the simulation-based definition offers a stronger form of security than the indistinguishability-based definition, in fact, this remains unknown. It does seem, however, that protocols satisfying SEM-SO-ENC security are harder to construct.

In [BHY09] it was shown that if a lossy encryption scheme has an efficient algorithm opener that can "open" a lossy ciphertext to a desired plaintext, then the scheme is already SEM-SO-ENC secure. Since our constructions of Re-randomizable encryptions give rise to lossy encryptions, to create SEM-SO-ENC security from known assumptions, it suffices to check which re-randomizable encryptions have an efficient opener algorithm.

When we instantiate our encryption scheme with the Paillier Cryptosystem, or the Goldwasser-Micali cryptosystem, the factorization of the modulus N allows us to devise an efficient opening algorithm. That the Goldwasser-Micali scheme is SEM-SO-ENC secure was already recognized in [BHY09], however instantiating our re-randomizable encryption with the Paillier Cryptosystem gives rise to the first SEM-SO-ENC secure cryptosystem under the Decisional Composite Residuosity (DCR) assumption.

13.1 Simulation-Based Security

While we have mostly focused on an indistinguishability-based notion of security under selecting openings, in [BHY09], Hofheinz et al. also formalized a simulation-based notion of security under selective openings. Their simulation-based definition of security intuitively seems stronger than the indistinguishability-based definition, however, it remains unknown whether SEM-SO-ENC implies IND-SO-ENC.

Definition 5. (Semantic Security under selective openings/SEM-SO-ENC).

Let Enc be a Public Key Cryptosystem (PKC), we say that Enc is simulatable under selective openings (SEM-SO-ENC secure) if for every PPT message distribution M, every PPT adversary A, and every PPT relation $\mathcal{R}$, there exists an efficient simulator $S = (S_1, S_2)$ such that we have that

$$\left|\Pr\left[A^{\mathsf{sem\text{-}so\text{-}real}} = 1\right] - \Pr\left[A^{\mathsf{sem\text{-}so\text{-}ideal}} = 1\right]\right| < \nu$$

for some negligible function ν, and where the games $\textsf{sem-so-real}$ and $\textsf{sem-so-ideal}$ are defined as follows

SEM-SO-ENC (Real)	SEM-SO-ENC (Ideal)
• $(m_1, \ldots, m_n) \leftarrow M$ • $r_1, \ldots, r_n \leftarrow \textsf{coins}(E)$ • $I \leftarrow A((E(m_1, r_i), \ldots, E(m_n, r_n))$ • $w \leftarrow A(((m_i, r_i))_{i \in I})$ • Output $\mathcal{R}(m, w)$.	• $(m_1, \ldots, m_n) \leftarrow M$. • $(I, st) \leftarrow S_1(1^\lambda)$. • $w \leftarrow S_2(st, \{m_i\}_{i \in I})$. • Output $\mathcal{R}(m, w)$.

More explicitly, in the real game,

- The challenger samples messages $(m_1, \ldots, m_n) \leftarrow M$, from the joint message distribution.
- The challenger generates randomness $r_1, \ldots, r_n \leftarrow \textsf{coins}(E)$.
- The challenger sends $(E(m_1, r_1), \ldots, E(m_n, r_n)$ to A.
- The adversary A responds with a subset $I \subset \{1, \ldots, n\}$, with $|I| = n/2$.
- The challenger reveals *both* m_i and r_i for $i \in I$.
- The adversary outputs a string w.
- The value of the game is $\mathcal{R}(m, w)$.

In the ideal game,

- The challenger samples messages $(m_1, \ldots, m_n) \leftarrow M$, from the joint message distribution.
- Without seeing any encryptions, the simulator chooses a subset I, and some state information st.
- Without seeing any randomness, after seeing the messages $\{m_i\}_{i \in I}$, and the state information, the simulator outputs a string w.
- The value of the game is $\mathcal{R}(m, w)$.

In [BHY09], Hofheinz, Bellare and Yilek, proved that a lossy encryption scheme, with an *efficient* $\textsf{opener}$ procedure are SEM-SO-ENC secure.

Definition 6. A *lossy public-key encryption scheme with efficient opening* is a tuple $(G_{\textsf{inj}}, G_{\textsf{lossy}}, E, D)$ satisfying Definition 4, with the additional property that the algorithm $\textsf{opener}$ is efficient, i.e.

- *Openability.* There is an *efficient* algorithm $\textsf{opener}$, such that if $(pk_{\textsf{lossy}}, sk_{\textsf{lossy}}) \leftarrow G_{\textsf{lossy}}$, and $r \leftarrow \textsf{coins}(E)$, then for all $x_0, x_1 \in X$ with all but negligible probability, $r' \leftarrow \textsf{opener}(pk_{\textsf{lossy}}, E(pk_{\textsf{lossy}}, x_0, r))$, and $E(pk_{\textsf{lossy}}, x_1, r')$.

Theorem 3. Lossy Encryption with efficient opening is SEM-SO-ENC secure.

Proof. This is Theorem 2 in [BHY09]. The proof is straightforward, and we only sketch it here.
We proceed in a series of games.

- G_0 is the real SEM-SO-ENC experiment.
- G_1 is the same as G_0, except the adversary is given a lossy public key, instead of a real public key.
- G_2 instead of giving the adversary the real randomness $\{r_i\}_{i \in I}$, the Challenger uses the efficient opener procedure to generate valid randomness.
- G_3 instead of giving the adversary encryptions of m_i, the adversary is given encryptions of a dummy message δ, but the adversary is still given openings to actual messages $\{m_i\}_{i \in I}$ obtained from the opener procedure.

Now, the simulator can simulate G_3 with the adversary. The simulator generates a lossy key pair, and encrypts a sequence of dummy messages and forwards the encryptions to A. The adversary, A, replies with a set I, which S forwards to the challenger. Then S uses the efficient opener procedure to open the selected messages for A. At which point A outputs a string w, and S outputs the same string. Since the outputs of A in G_0 and G_3 are computationally close, the outputs of S, and A in the real and ideal experiments will also be computationally close. □

13.2 Selective Opening Security From the Decisional Composite Residuosity Assumption

Here we give an overview of our construction when applied to the Paillier Cryptosystem (a review of the details of the Paillier Cryptosystem can be found in Appendix A).

By defining $\mathsf{ReRand}(c, r) = c \cdot E(pk, 0, r) \mod n^2$, we obtain IND-SO-COM secure commitments and IND-SO-ENC secure encryptions through our general construction in §9.

It was already known how to build IND-SO-ENC from DCR, since Peikert and Waters [PW08], and Boldyreva, Fehr and O'Neill showed how to build Lossy-Trapdoor Functions from DCR, and Bellare, Hofheinz and Yilek showed that Lossy-Trapdoor Functions imply IND-SO secure encryptions. We note, however, that our constructions are significantly more efficient than those that follow from [PW08], and somewhat more efficient than those that follow from [BFO08].

While the results of [BHY09] imply that IND-SO-ENC secure encryptions follow from DCR, the question of SEM-SO-ENC secure encryptions was left open, indeed, the only previous construction of SEM-SO-ENC secure encryptions were given in [BHY09] and based off of the Quadratic Residuosity Assumption (QR). By instantiating our scheme in 9 with the Paillier (or Damgård-Jurik) cryptosystem, we observe that the function opener is efficient, and hence the results of [BHY09] show that the resulting encryption scheme achieves SEM-SO-ENC security.

To see this, recall that $E(pk, m, r) = c^m r^N \mod N^2$, where c is an Nth power in lossy mode. Thus, the algorithm opener, on input $e = r_1^N$ and some target message m must find $r' \in \mathbb{Z}/N\mathbb{Z}$ such that $c^m (r')^N = e$. If we write $c = r_0^N$, then opener must find a solution to

$$(r')^N = \left(\frac{r_1}{r_0^m}\right)^n.$$

So the efficiency of opener reduces to the efficiency of taking Nth roots modulo N^2. But this is easily done if the factorization of N is known, since we can set $d = N^{-1} \mod \phi(N)$, and then taking Nth roots, is equivalent to exponentiating modulo N, i.e.

$$(r^N)^d = r^{Nd} = r \mod N.$$

Thus we immediately get a SEM-SO-COM secure encryption protocol from the DCR assumption. Thus we arrive at

Corollary 6. Under the Decisional Composite Residuosity assumption (DCR), the system described in §12.1 is SEM-SO-ENC secure.

Since the Paillier cryptosystem (and the Damgård-Jurik extension, have smaller ciphertext expansion than the Goldwasser-Micali cryptosystem (which only encrypts bits), we arrive at more efficient system than the only known SEM-SO-ENC secure cryptosystem.

14 Chosen Ciphertext Security

14.1 Definition

We extend the notion of a Chosen Ciphertext Attack (CCA) ([NY90],[RS91],[DDN91]) to the selective opening setting.

We define two games, a real game ($\mathsf{ind\text{-}cca2\text{-}real}$) and an ideal game ($\mathsf{ind\text{-}cca2\text{-}ideal}$). In both games, the challenger runs the key-generation algorithm to generate a public-key secret-key pair, and sends the public-key to the adversary. The adversary is then allowed to adaptively make two types of queries.

- **Selective Opening Query:** The adversary A chooses a message distribution M, and sends a description of M to the challenger. The challenger samples $(m_1, \ldots, m_n) \leftarrow M$, and generates

$$(c_1, \ldots, c_n) = (E(pk, m_1, r_1), \ldots, E(pk, m_n, r_n)).$$

  The challenger sends $(c_1, \ldots, c_n)$ to the adversary, and the adversary chooses a subset $I \subset [n]$, with $|I| = n/2$, and sends I to the challenger. The challenger then sends $\{(m_i, r_i)\}_{i \in I}$ to the Adversary. We call the ciphertexts $c_1, \ldots, c_n$ *target ciphertexts*.

  - In the real game, the challenger then sends $\{m_j\}_{j \notin I}$ to the adversary.
  - In the ideal game, the challenger resamples $(m'_1, \ldots, m'_n) \leftarrow M|_{M_I}$, and sends $\{m'_j\}_{j \notin I}$ to the adversary.

- **Decryption Queries:** The adversary A chooses a ciphertext c that has never appeared as a target ciphertext, and sends c to the challenger. If c is a valid ciphertext (i.e. $D(c) \neq \perp$) then the challenger responds with $m = D(c)$.

After adaptively making polynomially many queries, with at most one of them being a Selective Opening Query, the adversary outputs a bit b.

Definition 7. (IND-SO-CCA2) A public key encryption scheme E is called IND-SO-CCA2 secure, if, for all PPT adversaries A, A's output in the real game is negligibly different from its output in the ideal game, i.e.

$$\left|\Pr[A^{\mathsf{ind\text{-}cca2\text{-}real}} = 1] - \Pr[A^{\mathsf{ind\text{-}cca2\text{-}ideal}} = 1]\right| < \nu.$$

For some negligible function ν.

We remark that if the adversary is not allowed to make decryption queries, this reduces to IND-SO-ENC security.

14.2 Non-Interactive Zero Knowledge

The most successful technique in constructing systems secure against an adaptive chosen ciphertext attack has been the Naor-Yung paradigm [NY90]. Roughly, the idea is to encrypt the message twice and include a Non-Interactive Zero Knowledge (NIZK) proof that both encryptions encrypt the same plaintext. The proof of security then uses the simulator for the NIZK to simulate the proof for the challenge ciphertext. This method has since been refined in [DDN91],[Sah99],[SCO+01], and [Lin06] (among others).

Our constructions of IND-SO-CCA2 encryption follow the general structure of the Naor-Yung paradigm [NY90], however, the selective opening of the encryption query poses new challenges. In particular, if we naïvely try to follow the Naor-Yung paradigm, we immediately encounter difficulties because our challenger must reveal the messages and randomness for half of the ciphertexts in the challenge. This will immediately reveal to the adversary that the proofs were simulated. It requires new ideas to overcome this difficulty.

We now give a brief definition of the properties of a Non-Interactive Zero Knowledge Proof of Knowledge with Honest Prover State-Reconstruction (originally defined and constructed in [GOS06]).

Let $\mathcal{R}$ be an efficiently computable binary relation and let $L = \{x : \exists w \text{ such that } (x, w) \in \mathcal{R}\}$. We refer to L as a language, x as a statement, and w as a witness.

A Non-Interactive Proof System for L is a triple of PPT algorithms $(\mathsf{CRSgen}, \mathsf{Prover}, \mathsf{Verifier})$ such that

- $\sigma \leftarrow \mathsf{CRSgen}(1^\lambda)$.
  Generates a Common Reference String.
- $\pi \leftarrow \mathsf{Prover}(CRS, x, w)$.
  On inputs x, and a witness w for x, such that $\mathcal{R}(x, w) = 1$, the Prover outputs a proof π.
- $b \leftarrow \mathsf{Verifier}(CRS, x, p)$.
  On inputs x and a purported proof π, $\mathsf{Verifier}$ outputs a bit b.

Definition 8. A triple of algorithms is called a Non-Interactive Zero Knowledge Proof of Knowledge if

- **Completeness:** For all adversaries A,
  $$\Pr\left[\sigma \leftarrow \mathsf{CRSgen}(1^\lambda); (x, w) \leftarrow A(\sigma); \pi \leftarrow \mathsf{Prover}(\sigma, x, w) : \mathsf{Verifier}(\sigma, x, \pi) = 1 \text{ if } (x, w) \in R\right] > 1-\nu,$$
  For some negligible function ν.
- **Soundness:** For all adversaries A,
  $$\Pr\left[\sigma \leftarrow \mathsf{CRSgen}(1^\lambda); (x, \pi) \leftarrow A(\sigma) : \mathsf{Verifier}(\sigma, x, \pi) = 0 \text{ if } x \notin L\right] > 1 - \nu.$$
- **Knowledge Extraction:** There exists an extractor $\mathsf{Ext} = (\mathsf{Ext}_1, \mathsf{Ext}_2)$ such that for all adversaries A
  $$\left|\Pr\left[\sigma \leftarrow \mathsf{CRSgen}(1^\lambda) : A(\sigma) = 1\right] - \Pr\left[(\sigma, \tau) \leftarrow \mathsf{Ext}_1(1^\lambda) : A(\sigma) = 1\right]\right| < \nu$$
  and
  $$\Pr\left[(\sigma, \tau) \leftarrow \mathsf{Ext}_1(1^\lambda); (x, \pi) \leftarrow A(\sigma); w \leftarrow \mathsf{Ext}_2(\sigma, \tau, x, \pi) : \mathsf{Verifier}(\sigma, x, \pi) = 0 \text{ or } (x, w) \in \mathcal{R}\right] > 1-\nu$$
  For some negligible function ν.
- **Zero-Knowledge:** There exists a simulator $S = (S_1, S_2)$, such that for all adversaries A,
  $$\left|\Pr\left[\sigma \leftarrow \mathsf{CRSgen}(1^\lambda) : A^{P(\sigma,\cdot,\cdot)}(\sigma) = 1\right] - \Pr\left[(\sigma, \tau) \leftarrow S_1(1^\lambda) : A^{S'(\sigma,\tau,\cdot,\cdot)}(\sigma) = 1\right]\right| < \nu,$$
  where S' is defined
  $$S' = \begin{cases} S_2(\sigma, \tau, x) & \text{if } (x, w) \in \mathcal{R}, \\ \perp & \text{otherwise.} \end{cases}$$
- **Honest-Prover State Reconstruction:** There exists a simulator $S = (S_1, S_2, S_2)$ such that for all adversaries A
  $$\left|\Pr\left[\sigma \leftarrow \mathsf{CRSgen}(1^\lambda); A^{PR(\sigma,\cdot,\cdot)}(\sigma) = 1\right] - \Pr\left[(\sigma, \tau) \leftarrow S_1(1^\lambda) : A^{SR(\sigma,\tau,\cdot,\cdot)}(\sigma) = 1\right]\right| < \nu,$$
  where $PR(\sigma, x, w)$ samples $r \leftarrow \mathsf{coins}(\mathsf{Prover})$, and sets $\pi = \mathsf{Prover}(\sigma, x, w, r)$, and returns (π, r), and where SR samples $r^* \leftarrow \mathsf{coins}(S_2)$, and sets $\pi' = S_2(\sigma, \tau, x, r')$, finally PR sets $r' \leftarrow S_3(\sigma, \tau, x, w, r^*)$ and returns (π', r'). Both oracles output $\perp$ if $(x, w) \notin \mathcal{R}$.

14.3 Strongly Unforgeable Signatures

A signature scheme is a triple of PPT algorithms $(\mathsf{G}, \mathsf{Sign}, \mathsf{Ver})$ such that

- The algorithm G takes a security parameter λ, and returns a verification key and a signing key.

$$(vk, sk) \leftarrow \mathsf{G}(1^\lambda).$$

- The algorithm Sign takes a message m and the signing key, and produces a signature sig.

$$\mathsf{sig} \leftarrow \mathsf{Sign}(m, sk).$$

- The algorithm Ver takes a verification key, a message, and a signature, and returns a bit b.

$$b \leftarrow \mathsf{Ver}(vk, m, \mathsf{sig}).$$

We require

- **Completeness:** For all m

$$\Pr\left[(vk, sk) \leftarrow \mathsf{G}; \mathsf{sig} \leftarrow \mathsf{Sign}(m, sk); \mathsf{Ver}(vk, m, \mathsf{sig}) = 1\right] = 1.$$

- **Strongly Unforgeable:** For all PPT adversaries A,

$$\Pr\Big[(vk, sk) \leftarrow \mathsf{G};\ (m, \mathsf{sig}') \leftarrow A^{\mathsf{Sign}(\cdot, sk)}(vk) : \\ \mathsf{Ver}(vk, m, \mathsf{sig}') = 1 \text{ and } \mathsf{sig}' \text{ was never the output of } \mathsf{Sign}(\cdot, sk)\Big]$$

If we restrict A to make at most one oracle query to $\mathsf{Sign}(\cdot, sk)$ we say that $(\mathsf{G}, \mathsf{Sign}, \mathsf{Ver})$ is a one-time strongly unforgeable signature scheme.

14.4 Unduplicatable Set Selection

Unduplicatable set selection was used implicitly in [NY90] and [CIO98], and formalized in [Sah99]. We wish to create a mapping from $\mathfrak{g} : \{0,1\}^k \to B$ such that for all distinct $a^1, \ldots, a^n, a^{n+1} \in \{0,1\}^k$,

$$\mathfrak{g}(a^{n+1}) \not\subset \bigcup_{i=1}^{n} \mathfrak{g}(a^i).$$

In [Sah99], Sahai gives a simple general construction based on polynomials. Let $\ell = 2^{\lceil \log_2 2nk \rceil}$, so $\ell > 2nk$, and let $Y = \mathbb{F}_\ell \times \mathbb{F}_\ell$, then to each $a \in \{0,1\}^k$. Now, we may associate a polynomial

$$f_a(x) = a_0 + a_1 x + \cdots a_{k-1} x^{k-1} \in \mathbb{F}_\ell[x].$$

Then if we set

$$\mathfrak{g}(a) = \{(t, f_a(t)) : t \in \mathbb{F}_\ell\}.$$

Now, $|\mathfrak{g}(a)| = \ell$, and if $a \neq a'$, we have $|\mathfrak{g}(a) \cap \mathfrak{g}(a')| \leq k - 1$. Thus

$$\begin{aligned}
\left|\mathfrak{g}(a^{n+1}) \setminus \bigcup_{i=1}^{n} \mathfrak{g}(a^i)\right| &= \left|\mathfrak{g}(a^{n+1}) \setminus \bigcup_{i=1}^{n} \mathfrak{g}(a^{n+1}) \cap \mathfrak{g}(a^i)\right| \\
&\geq \left|\mathfrak{g}(a^{n+1})\right| - \sum_{i=1}^{n} \left|\mathfrak{g}(a^{n+1}) - \cap \mathfrak{g}(a^i)\right| \\
&\geq \ell - n(k-1) \\
&\geq \frac{\ell}{2}.
\end{aligned}$$

We call $\mathfrak{g}$ an (n, k, ℓ) unduplicatable set selector.

14.5 An IND-SO-CCA2 Construction

Along with the NIZK Proofs of Knowledge with Honest Prover State Reconstruction, Our construction relies on a number of common cryptographic tools. We will also require a strongly unforgeable one-time signature scheme. In our game, a single encryption query is actually n separate encryptions, so we will require an unduplicatable set selector $\mathfrak{g}$ for sets of size n. Finally, we will require an IND-SO-ENC secure cryptosystem.

Let (G_{so}, E, D) be an IND-SO-ENC secure cryptosystem. Let $(\mathsf{G}, \mathsf{Sign}, \mathsf{Ver})$ be a strongly unforgeable one-time signature scheme. Let $\mathfrak{g}$ be an (n, λ) unduplicatable set selector, and let $\ell = |g(0^\lambda)|$, and $L = \mathfrak{g}(\{0,1\}^\lambda)$.

Let $(\mathsf{CRSgen}, \mathsf{Prover}, \mathsf{Verifier})$ be a Noninteractive Zero Knowledge Proof of Knowledge with Honest Prover State Reconstruction for the language given by the relation $((e_0, e_1), (m, r_0, r_1)) \in \mathcal{R}$ if $e_0 = E(m, r_0)$ and $e_1 = E(m, r_1)$.

Our scheme works as follows

- **KeyGen:**

$$(pk_0, sk_0) \leftarrow G_{so}(1^\lambda),\ (pk_1, sk_1) \leftarrow G_{so}(1^\lambda),\ \text{and } (\sigma_i, \tau_i) \leftarrow \mathsf{Ext}_1(1^\lambda) \text{ for } i \in L$$

  Set

$$pk = (vk, pk_0, pk_1, \{\sigma_i\}_{i \in L}) \quad \text{and} \quad sk = (sk_0, sk_1, \{\tau_i\}_{i \in L}).$$

- **Encryption:** Pick

$$r^{sig} \leftarrow \mathsf{coins}(\mathsf{Sign}),\ r_0 \leftarrow \mathsf{coins}(E),\ r_1 \leftarrow \mathsf{coins}(E),\ r_i^{nizk} \leftarrow \mathsf{coins}(\mathsf{Prover}) \text{ for } i = 1, \ldots, \ell.$$

  Generate keys for a one-time signature using randomness r^{sig}.

$$(vk, sk) = \mathsf{G}(r^{sig}).$$

  For a message m, calculate

$$e_0 = E(pk_0, m, r_0), \quad e_1 = E(pk_1, m, r_1)$$

  set $w = (m, r_0, r_1)$.

$$\overline{\pi} = (\pi_1, \ldots, \pi_\ell) = (\mathsf{Prover}(\sigma_i, (e_0, e_1), w))_{i \in \mathfrak{g}(vk)}$$

  using randomness r_i^{nizk} in the ith iteration of Prover. set

$$\mathsf{sig} = \mathsf{Sign}(e_0, e_1, \overline{\pi}),$$

  output the ciphertext

$$c = (vk, e_0, e_1, \overline{\pi}, \mathsf{sig})$$

- **Decryption:** Given a ciphertext

$$c = (vk, e_0, e_1, \overline{\pi}, \mathsf{sig})$$

  Check that

$$\mathsf{Ver}(vk, (e_0, e_1, \overline{\pi})) = 1,$$

  otherwise return $\perp$. For $i \in \mathfrak{g}(vk)$, check that

$$\mathsf{Verifier}(\sigma_i, (e_0, e_1), \pi_i) = 1,$$

  otherwise return $\perp$.

  Pick a random $i \in \mathfrak{g}(vk)$ and use the Extractor Ext_2 to recover the witness (m, r_0, r_1), i.e.

$$(m, r_0, r_1) \leftarrow \mathsf{Ext}_2(\sigma_i, \tau_i, (e_0, e_1), \pi_i)$$

  return m.

Theorem 4. This scheme is IND-SO-CCA2 secure.

Proof. We consider a series of games

- Game 0: This is the $\mathsf{ind\text{-}cca2\text{-}real}$ game.
- Game 1: Pick the verification key $((vk^{chal,1}, sk^{chal,1}), \dots, (vk^{chal,n}, sk^{chal,n}))$ to be used in the challenge ciphertexts during parameter generation.
- Game 2: Generate the Common Reference Strings by

$$\sigma_i = \begin{cases} S_1(1^\lambda) & \text{if } i \in \mathfrak{g}(vk^{chal,j}) \text{ for some } j \in [n] \\ \mathsf{Ext}_1(1^\lambda) & \text{otherwise.} \end{cases}$$

  In decryption, we now use $i \notin \mathfrak{g}(g(vk))$ to recover (m, r_0, r_1).

- Game 3: When generating the target ciphertexts, ignore the witness and generate the "proof"

$$\overline{\pi} = \{\pi_i\}_{i \in \mathfrak{g}(vk)} = \{S_2(\sigma_i, \tau_i(e_0, e_1), r_i^*\}_{i \in \mathfrak{g}(vk)}$$

  when the adversary asks for the decryption and randomness of a subset of the target ciphertexts, use the State Reconstructor to generate

$$r_i \leftarrow S_3(\sigma_i, \tau_i, (e_0, e_1), (m, r_0, r_1, r_i^*)),$$

  and return these r_i instead of the r_i^* that were actually used.

Let W_i be the event that the adversary wins game i. Clearly $W_0 = W_1$, since from the adversary's point of view they are identical. To show that W_1 and W_2 are only negligibly different, notice that by the strong unforgeability of $(\mathsf{G}, \mathsf{Sign}, \mathsf{Ver})$, the adversary can never ask for the decryption of a ciphertext signed with vk, so by the unduplicatability of $\mathfrak{g}$, there will always be at least one valid proof generated with an extractable CRS. Now, it's easy to see that any PPT adversary that can distinguish between Game 2 and Game 1 can be used to distinguish the CRS generated by the extraction simulator Ext_1, and Honest Prover Reconstruction simulator S_1 (really $n\ell$ such simulators), but if

$$\left| \Pr\left[(\sigma, \tau) \leftarrow S_1(1^\lambda) : A^{SR(\sigma,\tau,\cdot,\cdot)}(\sigma) = 1 \right] - \Pr\left[(\sigma, \tau) \leftarrow S_1(1^\lambda) : A^{S'(\sigma,\tau,\cdot,\cdot)}(\sigma) = 1 \right] \right| > \epsilon,$$

the either

$$\left| \Pr\left[\sigma \leftarrow \mathsf{CRSgen}(1^\lambda) : A^{P(\sigma,\cdot,\cdot)}(\sigma) = 1 \right] - \Pr\left[(\sigma, \tau) \leftarrow S_1(1^\lambda) : A^{S'(\sigma,\tau,\cdot,\cdot)}(\sigma) = 1 \right] \right| > \frac{\epsilon}{2},$$

or

$$\left| \Pr\left[\sigma \leftarrow \mathsf{CRSgen}(1^\lambda); A^{PR(\sigma,\cdot,\cdot)}(\sigma) = 1 \right] - \Pr\left[(\sigma, \tau) \leftarrow S_1(1^\lambda) : A^{SR(\sigma,\tau,\cdot,\cdot)}(\sigma) = 1 \right] \right| > \frac{\epsilon}{2}.$$

Since these are both negligible by the definition of our NIZK, the difference between W_1 and W_2 is negligible.

To see that the difference between W_2 and W_3 is negligible, we notice that we can immediately transform an adversary that distinguishes Game 2 from Game 3 into an adversary that breaks the indistinguishability of the Honest Prover State Reconstruction simulator, losing a factor of $n\ell$ (because we are making $n\ell$ comparisons).

To complete the proof, we must show that $|W_3 - \frac{1}{2}|$ is negligible. Towards a contradiction suppose $|W_3 - \frac{1}{2}| > \epsilon$. We will transform the adversary A into an adversary that breaks the IND-SO-ENC security of (G_{so}, E, D). To see this, notice that in Game 3, we never use the decryption keys for the underlying IND-SO-ENC encryption, and we do not need the randomness used for the challenge ciphertext until *after* the adversary has requested I. So given 2 public-keys for the underlying (G_{so}, E, D), we break the IND-SO-ENC security as follows. We generate the corresponding NIZKs, OWTDPs and signatures according to Game 3, and play the IND-SO-CCA2 game with the adversary. When the adversary makes an encryption query, we forward the distribution M to the IND-SO-ENC challenger who responds with two sets of ciphertexts $(e_0^1, \ldots, e_0^n)$ and $(e_1^1, \ldots, e_1^n)$. Using the S simulator, we wrap these into a challenge ciphertext in the CCA2 scheme and forward this to the adversary. When the adversary responds with a set I, we forward this set I to the IND-SO-ENC challenger and receive decryptions and randomness. Using the S simulator, we can use the witnesses provided by IND-SO-ENC challenger to generate the necessary randomness for the NIZKs in the IND-SO-CCA2 challenge ciphertexts. Finally, when A outputs a bit b, we output the same bit. Since A is correct with probability $\frac{1}{2}+\epsilon$, break the IND-SO-ENC security of (G_{so}, E, D) with probability $\frac{1}{2}+\epsilon$.

□

15 Conclusion

We have shown that re-randomizable encryption implies perfectly-binding IND-SO-COM secure commitments, and IND-SO-ENC secure encryptions. In the process we have shown that re-randomizable encryption implies Lossy Encryption, which is interesting in its own right. Both constructions are relatively simple and retain the efficiency of the underlying re-randomizable encryption protocol. Our constructions can be applied to known cryptosystems, and immediately yields simple and efficient IND-SO-COM secure commitments and IND-SO-ENC secure encryptions from the Decisional Diffie-Hellman (DDH), Decisional-Composite Residuosity (DCR) and Quadratic Residuosity (QR) assumptions.

We have shown a black-box construction of perfectly binding IND-SO-COM commitments from Re-randomizable Encryption, perhaps a surprising result in light of the black box separations given in [BHY09]. Our results can be applied to create efficient perfectly binding IND-SO-COM commitments from the DDH, DCR and QR assumptions.

Applying our general construction to the Paillier Cryptosystem yields the first construction of SEM-SO-ENC secure encryptions from the DCR assumption, and this construction is the most efficient that is currently known.

We formalized Chosen Ciphertext security in the selective opening setting, and gave a general construction based existing primitives.

References

[Bea97] Donald Beaver. Plug and play encryption. In *CRYPTO '97*, pages 75–89, London, UK, 1997. Springer-Verlag.

[BFO08] Alexandra Boldyreva, Serge Fehr, and Adam O'Neill. On notions of security for deterministic encryption, and efficient constructions without random oracles. In David Wagner, editor, *CRYPTO '08*, volume 5157 of *Lecture Notes in Computer Science*, pages 335–359. Springer, 2008.

[BH92] Donald Beaver and Stuart Haber. cryptographic protocols provably secure against dynamic adversaries. In *EUROCRYPT '92*, number 658 in Lecture Notes in Computer Science, pages 307–323. Springer-Verlag, 1992.

[BHY09] Mihir Bellare, Dennis Hofheinz, and Scott Yilek. Possibility and impossibility results for encryption and commitment secure under selective opening. In *EUROCRYPT '09*. Springer, 2009. Preprint received from the authors 1-29-09.

[CDNO97] Ran Canetti, Cynthia Dwork, Moni Naor, and Rafail Ostrovsky. Deniable encryption. In *CRYPTO '97: Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology*, pages 90–104, London, UK, 1997. Springer-Verlag.

[CFGN96] Ran Canetti, Uri Feige, Oded Goldreich, and Moni Naor. Adaptively secure multi-party computation. In *STOC '96: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 639–648, New York, NY, USA, 1996. ACM.

[CHK05] Ran Canetti, Shai Halevi, and Jon Katz. Adaptively-secure, non-interactive public-key encryption. In *TCC '05*, number 3378 in Lecture Notes in Computer Science, pages 150–168. Springer-Verlag, 2005.

[CIO98] Giovanni Di Crescenzo, Yuval Ishai, and Rafail Ostrovsky. Non-interactive and non-malleable commitment. In *STOC '98*. ACM, 1998.

[DDN91] Danny Dolev, Cynthia Dwork, and Moni Naor. Non-malleable cryptography. In *STOC '91*, pages 542–552, 1991.

[DJ01] Ivan Damgård and Mads Jurik. A generalisation, a simplification and some applications of paillier's probabilistic public-key system. In *PKC '01: Proceedings of the 4th International Workshop on Practice and Theory in Public Key Cryptography*, pages 119–136, London, UK, 2001. Springer-Verlag.

[DNRS03] Cynthia Dwork, Moni Naor, Omer Reingold, and Larry Stockmeyer. Magic functions: In memoriam: Bernard m. dwork 1923–1998. *Journal of the ACM*, 50(6):852–921, 2003.

[GOS06] Jens Groth, Rafail Ostrovsky, and Amit Sahai. Perfect non-interactive zero knowledge for np. In *Proceedings of Eurocrypt 2006, volume 4004 of LNCS*, pages 339–358. Springer, 2006.

[Gro04] Jens Groth. Rerandomizable and replayable adaptive chosen ciphertext attack secure cryptosystems. In *TCC '04*, volume 2951 of *Lecture Notes in Computer Science*, pages 152–170. Springer, 2004.

[Hof08] Dennis Hofheinz. Possibility and impossibility results for selective decommitments. Cryptology ePrint Archive, Report 2008/168, 2008. http://eprint.iacr.org/2008/168.

[JJS04] Markus Jakobsson, Ari Juels, and Paul Syverson. Universal re-encryption for mixnets. In *In Proceedings of the 2004 RSA Conference, Cryptographers track*, pages 163–178. Springer-Verlag, 2004.

[KN08] Gillat Kol and Moni Naor. Cryptography and game theory: Designing protocols for exchanging information. In *TCC '08 : Proceedings of the fifth annual Theory of Cryptography Conference*, pages 320–339. Springer Berlin / Heidelberg, 2008.

[Lin06] Yehuda Lindell. A simpler construction of cca2-secure public-key encryption under general assumptions. *Journal of Cryptology*, 19(3):359–377, 2006.

[NP01] Moni Naor and Benny Pinkas. Efficient oblivious transfer protocols. In *SODA '01*, pages 448–457. ACM/SIAM, 2001.

[NY90] Moni Naor and Moti Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *STOC '90*, pages 427–437, 1990.

[Pai99] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology – EUROCRYPT '99*, volume 1592 of *Lecture Notes in Computer Science*, pages 223–238. Springer Berlin / Heidelberg, 1999.

[PR07] Manoj Prabhakaran and Mike Rosulek. Rerandomizable RCCA encryption. In *CRYPTO '07*, volume 4622 of *Lecture Notes in Computer Science*, pages 517–534. Springer Berlin / Heidelberg, 2007.

[PVW08] Chris Peikert, Vinod Vaikuntanathan, and Brent Waters. A framework for efficient and composable oblivious transfer. In David Wagner, editor, *CRYPTO '08*, volume 5157 of *Lecture Notes in Computer Science*, pages 554–571. Springer, 2008.

[PW08] Chris Peikert and Brent Waters. Lossy trapdoor functions and their applications. In *STOC '08: Proceedings of the 40th annual ACM symposium on Theory of computing*, pages 187–196, New York, NY, USA, 2008. ACM.

[RS91] Charles Rackoff and Daniel Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack". In *CRYPTO '91*, pages 433–444, 1991.

[Sah99] Amit Sahai. Non-malleable non-interactive zero-knowledge, and adaptive chosen-ciphertext security. In *FOCS '99*, pages 543–553, 1999.

[SCO+01] Alfredo De Santis, Giovanni Di Crescenzo, Rafail Ostrovsky, Giuseppe Persiano, and Amit Sahai. Robust non-interactive zero knowledge. In *CRYPTO '01*, volume 2139 of *Lecture Notes in Computer Science*, pages 566–598. Springer Berlin / Heidelberg, 2001.

Appendix

A The Paillier Cryptosystem

We briefly review the Paillier Cryptosystem proposed by Pascal Paillier in [Pai99], and extended by Damgård and Jurik in [DJ01].

The cryptosystem works in $(\mathbb{Z}/N^2\mathbb{Z})^*$. From the Binomial Theorem, we have

$$(1+N)^a = 1 + aN \mod N^2,$$

so $(1+N)$ generates a cyclic subgroup of order N. In this group, we can take Discrete Logs efficiently by $L(x) = \frac{x-1}{N}$, since

$$L((1+N)^a) = L(1+aN) = a.$$

Now, if g generates $\langle 1+N \rangle$, and $c = g^a$, then as with traditional logs

$$a = \frac{L(c)}{L(g)}.$$

Now, we are ready to describe Paillier's Cryptosystem

- **Parameter Generation:**

  - Generates primes p, q of length $\lambda/2$ and sets $N = pq$.
  - Generate $g \in \mathbb{Z}/N^2\mathbb{Z}$ such that N divides the order of g.
    This condition is easy to verify if you have the factorization of N.

  The public parameters are $pk = (N, g)$
  The secret key is $sk = \text{lcm}(p-1, q-1)$.

- **Encryption:**

  - $r \leftarrow \mathbb{Z}/N\mathbb{Z}$,
    (really you want to generate $r \in (\mathbb{Z}/N\mathbb{Z})^*$, but the distributions are statistically close).
  - For $m \in \mathbb{Z}/N\mathbb{Z}$,
    $E(pk, m, r) = g^m r^N \mod N^2$.

- **Decryption:**
  Given a ciphertext $c \in (\mathbb{Z}/N^2\mathbb{Z})^*$,

$$m = \frac{L(c^{sk}) \mod N}{L(g^{sk})} \mod N.$$

This cryptosystem is IND-CPA secure under the Decisional Composite Residuosity Assumption (DCR), which (informally) says

Assumption 1. Decisional Composite Residuosity/(DCR): If N is an λ-bit RSA modulus, (i.e. $N = pq$), then

$$\{g \leftarrow \mathbb{Z}/N^2\mathbb{Z}; g\} \approx_c \{g \leftarrow \mathbb{Z}/N^2\mathbb{Z}; g^N\}.$$