Possibility and Impossibility Results for Encryption and Commitment Secure under Selective Opening

Mihir Bellare[1], Dennis Hofheinz[2], and Scott Yilek[1]

[1] Dept. of Computer Science & Engineering, University of California at San Diego, 9500 Gilman Drive, La Jolla, CA 92093, USA
{mihir,syilek}@cs.ucsd.edu
http://www-cse.ucsd.edu/users/{mihir,syilek}
[2] CWI, Amsterdam
Dennis.Hofheinz@cwi.nl
http://www.cwi.nl/~hofheinz

Abstract. The existence of encryption and commitment schemes secure under selective opening attack (SOA) has remained open despite considerable interest and attention. We provide the first public key encryption schemes secure against sender corruptions in this setting. The underlying tool is lossy encryption. We then show that no non-interactive or perfectly binding commitment schemes can be proven secure with black-box reductions to standard computational assumptions, but any statistically hiding commitment scheme is secure. Our work thus shows that the situation for encryption schemes is very different from the one for commitment schemes.

1 Introduction

IND-CPA and IND-CCA are generally viewed as strong notions of encryption security that suffice for applications. However, there is an important setting where these standard notions do not in fact imply security and the search for solutions continues, namely, in the presence of selective-opening attack (SOA) [22, 13, 38, 18, 16, 14]. Let us provide some background on SOA and then discuss our results for encryption and commitment.

1.1 Background

THE PROBLEM. Suppose a receiver with public encryption key pk receives a vector $\mathbf{c} = (\mathbf{c}[1], \ldots, \mathbf{c}[n])$ of ciphertexts, where sender i created ciphertext $\mathbf{c}[i] = \mathcal{E}(pk, \mathbf{m}[i]; \mathbf{r}[i])$ by encrypting a message $\mathbf{m}[i]$ under pk and coins $\mathbf{r}[i]$ $(1 \leq i \leq n)$. It is important here that *the messages* $\mathbf{m}[1], \ldots, \mathbf{m}[n]$ *might be related*, but *the coins* $\mathbf{r}[1], \ldots, \mathbf{r}[n]$ *are random and independent*. Now, the adversary, given $\mathbf{c}$, is allowed to corrupt some size t subset $I \subseteq \{1, \ldots, n\}$ of senders (say $t = n/2$), obtaining not only their messages but *also their coins*, so that

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-3-642-01001-9_35

A. Joux (Ed.): EUROCRYPT 2009, LNCS 5479, pp. 1–35, 2009.

it has $\mathbf{m}[i], \mathbf{r}[i]$ for all $i \in I$. This is called a selective opening attack (SOA). The security requirement is that the privacy of the unopened messages, namely $\mathbf{m}[i_1], \ldots, \mathbf{m}[i_{n-t}]$ where $\{i_1, \ldots, i_{n-t}\} = \{1, \ldots, n\} \setminus I$, is preserved. (Meaning the adversary learns nothing more about the unopened messages than it could predict given the opened messages and knowledge of the message distribution. Formal definitions to capture this will be discussed later.) The question is whether SOA-secure encryption schemes exist.

Status and motivation. One's first impression would be that a simple hybrid argument would show that any IND-CPA scheme is SOA-secure. Nobody has yet been able to push such an argument through. (And, today, regarding whether IND-CPA implies SOA-security we have neither a proof nor a counterexample.) Next one might think that IND-CCA, at least, would suffice, but even this is not known. The difficulty of the problem is well understood and documented [22, 13, 16, 38, 18, 14], and whether or not SOA-secure schemes exist remains open.

Very roughly, the difficulties come from a combination of two factors. The first is that it is the random coins underlying the encryption, not just the messages, that are revealed. The second is that the messages can be related.

We clarify that the problem becomes moot if senders can erase their randomness after encryption, but it is well understood that true and reliable erasure is difficult on a real system. We will only be interested in solutions that avoid erasures.

The problem first arose in the context of multiparty computation, where it is standard to assume secure communication channels between parties [8, 17]. But, how are these to be implemented? Presumably, via encryption. But due to the fact that parties can be corrupted, the encryption would need to be SOA-secure. We contend, however, that there are important practical motivations as well. For example, suppose a server has SSL connections with a large number of clients. Suppose a virus corrupts some fraction of the clients, thereby exposing the randomness underlying their encryptions. Are the encryptions of the uncorrupted clients secure?

Commitment. Notice that possession of the coins allows the adversary to verify that the opening is correct, since it can compute $\mathcal{E}(pk, \mathbf{m}[i]; \mathbf{r}[i])$ and check that this equals $\mathbf{c}[i]$ for all $i \in I$. This apparent commitment property has been viewed as the core technical difficulty in obtaining a proof. The view that commitment is in this way at the heart of the problem has led researchers to formulate and focus on the problem of commitment secure against SOA [22]. Here, think of the algorithm $\mathcal{E}$ in our description above as the commitment algorithm of a commitment scheme, with the public key being the empty string. The question is then exactly the same. More generally the commitment scheme could be interactive or have a setup phase.

Independently of the encryption setting, selective openings of commitments commonly arise in zero-knowledge proofs. Namely, often an honest verifier may request that the prover opens a subset of a number of previously made commitments. Thus, SOA-security naturally becomes an issue here, particularly when considering the concurrent composition of zero-knowledge proofs (since then,

overall more openings from a larger set of commitments may be requested). The security of the unopened commitments is crucial for the zero-knowledge property of such a protocol, and this is exactly what SOA-security of the commitments would guarantee.

Definitions. Previous work [22] has introduced and used a semantic-style security formalization of security under SOA. A contribution of our paper is to provide an alternative indistinguishability-based formalization that we denote IND-SO-ENC for encryption and IND-SO-COM for commitment. We will also refer to semantic security formalizations SEM-SO-ENC and SEM-SO-COM.

1.2 Results for Encryption

We provide the first public-key encryption schemes provably secure against selective-opening attack. The schemes have short keys. (Public and secret keys of a fixed length suffice for encrypting an arbitrary number of messages.) The schemes are stateless and noninteractive, and security does not rely on erasures. The schemes are without random oracles, proven secure under standard assumptions, and even efficient. We are able to meet both the indistinguishability (IND-SO-ENC) and the semantic security (SEM-SO-ENC) definitions, although under different assumptions.

Closer look. The main tool (that we define and employ) is lossy encryption, an encryption analogue of lossy trapdoor functions [40] that is closely related to meaningful-meaningless encryption [34] and dual-mode encryption [41]. We provide an efficient implementation of lossy encryption based on DDH. We also show that any (sufficiently) lossy trapdoor function yields lossy encryption, thereby obtaining several other lossy encryption schemes via the lossy trapdoor constructions of [40, 10, 45].

We then show that any lossy encryption scheme is IND-SO-ENC secure, thereby obtaining numerous IND-SO-ENC secure schemes. If the lossy encryption scheme has an additional property that we call efficient openability, we show that it is also SEM-SO-ENC secure. We observe that the classical quadratic residuosity-based encryption scheme of Goldwasser and Micali [27] is lossy with efficient openability, thereby obtaining SEM-SO-ENC secure encryption. It is interesting in this regard that the solution to a long-standing open problem is a scheme that has been known for 25 years. (Only the proof was missing until now.)

Previous work. In the version of the problem that we consider, there is one receiver and many senders. Senders may be corrupted, with the corruption exposing their randomness and message. An alternative version of the problem considers a single sender and many receivers, each receiver having its own public and secret key. Receivers may be corrupted, with corruption exposing their secret key. Previous work has mostly focused on the receiver corruption version of the problem. Canetti, Feige, Goldreich and Naor [13] introduce and implement non-committing encryption, which yields SOA-secure encryption in the receiver corruption setting. However, their scheme does not have short keys. (Both the

public and the secret key in their scheme are as long as the total number of message bits ever encrypted.) Furthermore, Nielsen [38] shows that this is necessary. Canetti, Halevi and Katz [16] provide SOA-secure encryption schemes for the receiver corruption setting with short public keys, but they make use of (limited) erasures. (They use a key-evolving system where, at the end of every day, the receiver's key is updated and the previous version of the key is securely erased.) In the symmetric setting, Panjwani [39] proves SOA-security against a limited class of attacks.

Our schemes do not suffer from any of the restrictions of previous ones. We have short public and secret keys, do not rely on erasures, and achieve strong notions of security.

A natural question is why our results do not contradict Nielsen's negative result saying that no noninteractive public key encryption scheme with short and fixed keys is SOA-secure without erasures for an unbounded number of messages [38]. The reason is that we consider sender corruptions as opposed to receiver corruptions.

Discussion. It has generally been thought that the two versions of the problem (sender or receiver corruptions) are of equal difficulty. The reason is that corruptions, in either case, allow the adversary to verify an opening and appear to create a commitment. (Either the randomness or the decryption key suffices to verify an opening.) Our work refutes this impression and shows that sender corruptions are easier to handle than receiver ones. Indeed, we can fully resolve the problem in the former case, while the latter case remains open. (Achieving a simulation-based notion for receiver corruptions is ruled out by [38] but achieving an indistinguishability-based notion may still be possible.)

1.3 Results for Commitment

Previous work. In the zero-knowledge (ZK) setting, Gennaro and Micali [24] notice a selective opening attack and circumvent it by adapting the distribution of the messages committed to. Similarly, a number of works (e.g.,Dolev et al. [21], Prabhakaran et al.[42] in the ZK context) use "cut-and-choose" techniques on committed values, which is a specific form of selective opening. These works can prove security by using specific properties of the distributions of the committed values (e.g., the fact that the unopened values, conditioned on the opened values, are still uniformly distributed). The first explicit treatment of SOA-secure commitment is by Dwork, Naor, Reingold, and Stockmeyer [22]. They formalized the problem and defined SEM-SO-COM. On the negative side, they showed that the existence of a one-shot (this means non-interactive and without setup assumptions) SEM-SO-COM-secure commitment scheme implied solutions to other well-known cryptographic problems, namely, three-round ZK and "magic functions." This is evidence that simulation-based one-shot SOA-secure commitment is difficult to achieve. In particular, from Goldreich and Krawczyk [26], it is known that three-round black-box zero-knowledge proof systems exist only for

languages in BPP.[1] On the positive side Dwork et al. showed that any statistically hiding chameleon commitment scheme is SOA-secure. (This scheme would not be one-shot, which is why this does not contradict their negative results.)

Results for SEM-SO-COM. On the negative side, we show that no one-shot or perfectly binding commitment scheme can be shown SEM-SO-COM-secure using black-box reductions to standard assumptions. Here, by a standard assumption, we mean any assumption that can be captured by a game between a challenger and an adversary. (A more formal definition will be given later.) Most (but not all) assumptions are of this form. On the positive side, we show, via non-black-box techniques, that there exists an interactive SEM-SO-COM-secure commitment scheme under the assumption that one-way permutations exist.

Results for IND-SO-COM. On the negative side, we show that no perfectly hiding commitment scheme (whether interactive or not) can be shown IND-SO-COM secure using black-box reductions to standard assumptions. On the positive side, we show that any statistically hiding commitment scheme is IND-SO-COM secure. (We note that a special case of this result was already implicit in the work of Bellare and Rogaway [6].)

Closer look. Technically, we derive black-box impossibility results in the style of Impagliazzo and Rudich [32], but we can derive stronger claims, similar to Dodis et al. [20]. (Dodis et al. [20] show that the security of full-domain hash signatures [4] cannot be proved using a black-box reduction to any hardness assumption that is satisfied by a random permutation.) Concretely, we prove impossibility of $\forall\exists$semi-black-box proofs from *any* computational assumption that can be formalized as an oracle $\mathcal{X}$ and a corresponding security property $\mathcal{P}$ (i.e., a game between a challenger and an adversary) which the oracle satisfies. For instance, to model one-way permutations, $\mathcal{X}$ could be a truly random permutation and $\mathcal{P}$ could be the one-way game in which a PPT adversary tries to invert a random image. We emphasize that, somewhat surprisingly, our impossibility claim holds even if $\mathcal{P}$ models SOA-security. In that case, however, a reduction will necessarily be non-black-box, see Section 9 for a discussion. Concurrently to and independently from our work, Haitner and Holenstein [28] developed a framework to prove impossibility of black-box reductions from *any* computational assumption. While their formalism is very similar to ours (e.g., their definition of a "cryptographic game" matches our definition of a "property"), they apply it to an entirely different problem, namely, encryption scheme security in the presence of key-dependent messages.

[1] "Black-box" means here that the ZK simulator uses only the (adverserial) verifier's next-message function in a black-box way to simulate an authentic interaction. Jumping ahead, we will show that in many cases SOA-secure commitment cannot be proved using a black-box reduction to a standard computational assumption. Both statements are negative, but orthogonal. Indeed, it is conceivable that a security reduction uses specific, non-black-box properties of the adversary (e.g., it is common in reductions to explicitly make use of the adversary's complexity bounds), but neither scheme nor reduction use specifics (like the code) of the underlying primitive.

Relation to the encryption results. An obvious question is why our results for encryption and commitment are not contradictive. The answer is that our SOA-secure encryption scheme does not give rise to a commitment scheme. Our commitment results do show that the SOA-security of an encryption scheme cannot be proved using a black-box reduction, *but only if encryption constitutes a commitment.* Because we consider SOA-security under *sender* corruptions in the encryption setting, this is not the case. (Recall that with sender corruptions, an encryption opening does not reveal the secret key, so the information-theoretic argument of Nielsen [38] that any encryption scheme is committing does not apply.)

1.4 History

This paper was formed by merging two Eurocrypt 2009 submissions which were accepted by the PC under the condition that they merge. One, by Bellare and Yilek, contained the results on encryption. (Sections 1.1,3,4,5.) The other, by Hofheinz, contained the results on commitment. (Sections 1.2,6,7,8,9.) Both papers had independently introduced the indistinguishability definition of SOA-security, the first for encryption and the second for commitment. Full versions of both papers are available as [7, 31].

2 Notation

For any integer n, let 1^n be its unary representation and let $[n]$ denote the set $\{1, \ldots, n\}$. We let $a \leftarrow b$ denote assignment to a the result of evaluating b. If b is simply a tuple of values of size m, we will write $(b_1, \ldots, b_m) \leftarrow b$ when we mean that b is parsed into b_1 to b_m. We let $a \leftarrow\!\!\$\ b$ denote choosing a value uniformly at random from random variable b and assigning it to a.

We say a function $\mu(n)$ is negligible if $\mu \in o(n^{-\omega(1)})$. We let $\mathsf{neg}(n)$ denote an arbitrary negligible function. If we say some $p(n) = \mathsf{poly}(n)$, we mean that there is some polynomial q such that for all sufficiently large n, $p(n) \leq q(n)$. The statistical distance between two random variable X and Y over common domain D is $\Delta(X, Y) = \frac{1}{2} \sum_{z \in D} |\Pr[X = z] - \Pr[Y = z]|$ and we say that two random variables X and Y are δ-close if their statistical distance is at most δ and if δ is negligible, we might say $X \equiv_s Y$.

We denote by ϵ the empty string. For any strings m_0 and m_1, let $m_0 \oplus m_1$ denote the bitwise xor of the two strings. We use boldface letters for vectors, and for any vector $\mathbf{m}$ of n messages and $i \in [n]$, let $\mathbf{m}[i]$ denote the ith message in $\mathbf{m}$. For a set $I \subseteq [n]$ of indices $i_1 < i_2 < \ldots < i_l$, let $\mathbf{m}[I] = (\mathbf{m}[i_1], \mathbf{m}[i_2], \ldots, \mathbf{m}[i_l])$. For any set I (resp. any vector $\mathbf{m}$)(resp. any string m), let $|I|$ (resp. $|\mathbf{m}|$) (resp. $|m|$) denote the size of the set (resp. length of the vector) (resp. length of the string).

All algorithms in this paper are randomized, unless otherwise specified as being deterministic. For any algorithm A, let $\mathsf{Coins}_A(x_1, x_2, \ldots)$ denote the set of possible coins A uses when run on inputs $x_1, x_2, \ldots$. Let $A(x_1, x_2, \ldots; r)$ denote running algorithm A on inputs $x_1, x_2, \ldots$ and with coins $r \in \mathsf{Coins}_A(x_1, x_2, \ldots)$. Then $A(x_1, x_2, \ldots)$ denotes the random variable $A(x_1, x_2, \ldots; r)$ with r chosen

uniformly at random from $\mathsf{Coins}_A(x_1, x_2, \ldots)$. When we say an algorithm is efficient, we mean that it runs in polynomial time in its first input; if the algorithm is randomized we might also say it runs in probabilistic polynomial time (PPT). An unbounded algorithm does not necessarily run in polynomial time.

3 Encryption Related Definitions

3.1 Encryption Schemes

A public-key encryption scheme $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ is a triple of PT algorithms. The (randomized) key generation algorithm $\mathcal{K}$ takes as input a security parameter 1^λ and outputs a public key/secret key pair (pk, sk). The (randomized) encryption algorithm $\mathcal{E}$ takes as input a public key pk and a message m and outputs a ciphertext c. The decryption algorithm takes as input a secret key sk and a ciphertext C and outputs either the decryption m of c, or $\perp$, denoting failure. We require the correctness condition that for all (pk, sk) generated by $\mathcal{K}$, and for all messages m, $\mathcal{D}(sk, \mathcal{E}(pk, m)) = m$. The standard notion of security for public-key encryption scheme is indistinguishability under chosen-plaintext attack ($\mathsf{ind\text{-}cpa}$).

3.2 Encryption Security under Selective Opening

We consider both indistinguishability-based and simulation-based definitions of security for encryption under selective opening which we call $\mathsf{ind\text{-}so\text{-}enc}$ and $\mathsf{sem\text{-}so\text{-}enc}$, respectively.

Indistinguishability-based. For any public-key encryption scheme $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, any message sampler $\mathcal{M}$, and any adversary $A = (A_1, A_2)$, we say the $\mathsf{ind\text{-}so\text{-}enc}$-advantage of A with respect to $\mathcal{M}$ is

$$\mathbf{Adv}^{\mathsf{ind\text{-}so\text{-}enc}}_{A,\mathcal{AE},\mathcal{M},n,t}(\lambda) = 2 \cdot \Pr[\mathbf{Exp}^{\mathsf{ind\text{-}so\text{-}enc}}_{A,\mathcal{AE},\mathcal{M},n,t}(\lambda)] - 1,$$

where the $\mathsf{ind\text{-}so\text{-}enc}$ security experiment is defined in Figure 1, and $\mathcal{M}|_{I,\mathbf{m}_0[I]}$ returns a random n-vector $\mathbf{m}_1$ according to $\mathcal{M}$, subject to $\mathbf{m}_1[I] = \mathbf{m}_0[I]$. In other words, $\mathcal{M}|_{I,\mathbf{m}_0[I]}$ denotes conditionally resampling from the message space subject to the constraint that the messages corresponding to indices in I are equal to $\mathbf{m}_0[I]$.

We say that a public-key encryption scheme $\mathcal{AE}$ is $\mathsf{ind\text{-}so\text{-}enc}$-secure if for any efficient message sampler $\mathcal{M}$ that supports efficient conditional resampling and for all efficient adversaries A, the $\mathsf{ind\text{-}so\text{-}enc}$-advantage of A with respect to $\mathcal{M}$ is negligible in the security parameter.

In words, the experiment proceeds as follows. The adversary is given a public key pk and n ciphertexts $\mathbf{c}$ encrypted under public key pk. The messages corresponding to the n ciphertexts come from the joint distribution $\mathcal{M}$. The adversary then specifies a set I of t ciphertexts and receives the randomness $\mathbf{r}[I]$ used to generate those ciphertexts in addition to a message vector $\mathbf{m}_b$ such that $\mathbf{m}_b[I]$ were the actual messages encrypted using $\mathbf{r}[I]$ and the rest of $\mathbf{m}_b$ depends

$\underline{\textbf{Experiment } \mathbf{Exp}^{\mathsf{ind\text{-}so\text{-}enc}}_{A,\mathcal{AE},\mathcal{M},n,t}(\lambda)}$
$\mathbf{m}_0 \leftarrow\!\!\$\ \mathcal{M}(1^\lambda);\ b \leftarrow\!\!\$\ \{0,1\};\ (pk, sk) \leftarrow\!\!\$\ \mathcal{K}(1^\lambda)$
For $i = 1, \ldots, n(\lambda)$ do
  $\mathbf{r}[i] \leftarrow\!\!\$\ \mathsf{Coins}_{\mathcal{E}}(pk, \mathbf{m}_0[i])$
  $\mathbf{c}[i] \leftarrow \mathcal{E}(pk, \mathbf{m}_0[i]; \mathbf{r}[i])$
$(I, \mathsf{st}) \leftarrow\!\!\$\ A_1(1^\lambda, pk, \mathbf{c})$
$\mathbf{m}_1 \leftarrow\!\!\$\ \mathcal{M}|_{I,\mathbf{m}_0[I]}$
$b' \leftarrow\!\!\$\ A_2(\mathsf{st}, \mathbf{r}[I], \mathbf{m}_b)$
Return $(b = b')$

Fig. 1. The IND-SO-ENC security experiment

on the bit b. If b, which the experiment chooses randomly, is 0, the rest of the messages in the vector are the actual messages used to create the ciphertexts $\mathbf{c}$ that were given to the adversary. If $b = 1$, the rest of the messages are instead resampled from $\mathcal{M}$, conditioned on I and $\mathbf{m}_b[I]$. The adversary must then try to guess the bit b.

The definition is a natural extension of ind-cpa to the selective decryption setting. Intuitively, the definition means that an adversary, after adaptively choosing to open some ciphertexts, cannot distinguish between the actual unopened messages and another set of messages that are equally likely given the opened messages that the adversary has seen.

Simulation-based. For any public-key encryption scheme $\mathcal{AE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, any message sampler $\mathcal{M}$, any relation R, any adversary $A = (A_1, A_2)$, and any simulator $S = (S_1, S_2)$, we say the sem-so-enc-advantage of A with respect to $\mathcal{M}$, R, and S is

$$\begin{aligned}\mathbf{Adv}^{\mathsf{ind\text{-}so\text{-}enc}}_{A,S,\mathcal{AE},\mathcal{M},\mathsf{R},n,t}(\lambda) = &\Pr[\mathbf{Exp}^{\mathsf{sem\text{-}so\text{-}enc\text{-}real}}_{A,\mathcal{AE},\mathcal{M},\mathsf{R},n,t}(\lambda) = 1] \\ &- \Pr[\mathbf{Exp}^{\mathsf{sem\text{-}so\text{-}enc\text{-}ideal}}_{S,\mathcal{AE},\mathcal{M},\mathsf{R},n,t}(\lambda) = 1]\end{aligned}$$

where the sem-so-enc security experiments are defined in Figure 2.

We say that a public-key encryption scheme $\mathcal{AE}$ is sem-so-enc-secure if for any efficient message sampler $\mathcal{M}$, any efficiently computable relation R, and any efficient adversary A, there exists an efficient simulator S such that the sem-so-enc-advantage of A with respect to $\mathcal{M}$, R, and S is negligible in the security parameter.

$\underline{\textbf{Experiment } \mathbf{Exp}^{\mathsf{sem\text{-}so\text{-}enc\text{-}real}}_{A,\mathcal{AE},\mathcal{M},\mathsf{R},n,t}(\lambda)}$	$\underline{\textbf{Experiment } \mathbf{Exp}^{\mathsf{sem\text{-}so\text{-}enc\text{-}ideal}}_{S,\mathcal{AE},\mathcal{M},\mathsf{R},n,t}(\lambda)}$
$\mathbf{m} \leftarrow\!\!\$\ \mathcal{M}(1^\lambda);\ (pk, sk) \leftarrow\!\!\$\ \mathcal{K}(1^\lambda)$	$\mathbf{m} \leftarrow\!\!\$\ \mathcal{M}(1^\lambda)$
For $i = 1, \ldots, n(\lambda)$ do	$(I, \mathsf{st}) \leftarrow\!\!\$\ S_1(1^\lambda)$
$\mathbf{r}[i] \leftarrow\!\!\$\ \mathsf{Coins}_{\mathcal{E}}(pk, \mathbf{m}[i])$	$w \leftarrow\!\!\$\ S_2(\mathsf{st}, \mathbf{m}[I])$
$\mathbf{c}[i] \leftarrow \mathcal{E}(pk, \mathbf{m}[i]; \mathbf{r}[i])$	Return $\mathsf{R}(\mathbf{m}, w)$
$(I, \mathsf{st}) \leftarrow\!\!\$\ A_1(1^\lambda, pk, \mathbf{c})$	
$w \leftarrow\!\!\$\ A_2(\mathsf{st}, \mathbf{r}[I], \mathbf{m}[I])$	
Return $\mathsf{R}(\mathbf{m}, w)$	

Fig. 2. The two security experiments for SEM-SO-ENC

In words, the experiments proceed as follows. In the sem-so-enc-real experiment, the adversary A is given a public key pk and n ciphertexts $\mathbf{c}$ encrypted under public key pk. The messages corresponding to the n ciphertexts come from the joint distribution $\mathcal{M}$. The adversary then specifies a set I of t ciphertexts and receives the messages $\mathbf{m}[I]$ and randomness $\mathbf{r}[I]$ used to generate those ciphertexts. The adversary then outputs a string w and the output of the experiment is $\mathsf{R}(\mathbf{m}, w)$, the relation applied to the message vector and adversary's output. In the sem-so-enc-ideal experiment, a vector $\mathbf{m}$ of messages is chosen and the simulator, given only the security parameter, chooses a set I. The simulator is then given $\mathbf{m}[I]$, the messages corresponding to the index set I. Finally, the simulator outputs a string w and the output of the experiment is $\mathsf{R}(\mathbf{m}, w)$.

4 Lossy Encryption

The main tool we use in our results is what we call a *Lossy Encryption Scheme*. Informally, a lossy encryption scheme is a public-key encryption scheme with a standard key generation algorithm (which produces 'real' keys) and a lossy key generation algorithm (which produces 'lossy' keys), such that encryptions with real keys are committing, while encryptions with lossy keys are not committing. Peikert, Vaikuntanathan, and Waters [41] called such lossy keys "messy keys", for *me*ssage lo*ssy*, while defining a related notion called Dual-Mode Encryption. The notion of Lossy Encryption is also similar to Meaningful/Meaningless Encryption [34], formalized by Kol and Naor.

More formally, a *lossy public-key encryption scheme* $\mathcal{AE} = (\mathcal{K}, \mathcal{K}_{\mathsf{loss}}, \mathcal{E}, \mathcal{D})$ is a tuple of PT algorithms defined as follows. The key generation algorithm $\mathcal{K}$ takes as input the security parameter 1^λ and outputs a keypair (pk, sk); we call public keys generated by $\mathcal{K}$ real public keys. The lossy key generation algorithm $\mathcal{K}_{\mathsf{loss}}$ takes as input the security parameter and outputs a keypair (pk, sk); we call such pk lossy public keys. The encryption algorithm $\mathcal{E}$ takes as input a public key pk (either from $\mathcal{K}$ or $\mathcal{K}_{\mathsf{loss}}$) and a message m and outputs a ciphertext c. The decryption algorithm takes as input a secret key sk and a ciphertext c and outputs either a message m, or $\perp$ in the case of failure. We require the following properties from $\mathcal{AE}$:

1. *Correctness on real keys.* For all $(pk, sk) \leftarrow\!\!\$\, \mathcal{K}$ it must be the case that $\mathcal{D}(sk, \mathcal{E}(pk, m)) = m$. In other words, when the real key generation algorithm is used, the standard public-key encryption correctness condition must hold.
2. *Indistinguishability of real keys from lossy keys.* No polynomial-time adversary can distinguish between the first outputs of $\mathcal{K}$ and $\mathcal{K}_{\mathsf{loss}}$. We call the advantage of an adversary A distinguishing between the two the lossy-key-advantage of A and take it to mean the obvious thing, i.e., the probability that A outputs 1 when given the first output of $\mathcal{K}$ is about the same as the probability it outputs 1 when given the first output of $\mathcal{K}_{\mathsf{loss}}$.
3. *Lossiness of encryption with lossy keys.* For any $(pk, sk) \leftarrow \mathcal{K}_{\mathsf{loss}}$ and two distinct messages m_0, m_1, it must be the case that $\mathcal{E}(pk, m_0) \equiv_s \mathcal{E}(pk, m_1)$. We say the advantage of an adversary A in distinguishing between the two

is the lossy-ind advantage of A and take it to mean the advantage of A in the standard ind-cpa game *when the public key pk in the ind-cpa game is lossy.* Notice that because the ciphertexts are *statistically* close, even an unbounded distinguisher will have low advantage. We sometimes call ciphertexts created with lossy public keys *lossy ciphertexts.*

4. *Possible to claim any plaintext.* There exists a (possibly unbounded) algorithm Opener that, given a lossy public key pk_{loss}, message m, and ciphertext $c = \mathcal{E}(pk_{\mathsf{loss}}, m)$, will output $r' \in_R \mathsf{Coins}_{\mathcal{E}}(pk_{\mathsf{loss}}, m)$ such that $\mathcal{E}(pk_{\mathsf{loss}}, m; r') = c$. In other words, Opener will find correctly distributed randomness to open a lossy ciphertext to the plaintext it encrypts. It then directly follows from the lossiness of encryption that with high probability the opener algorithm can successfully open *any* ciphertext to *any* plaintext.

We note that the fourth property is already implied by the first three properties; the canonical (inefficient) Opener algorithm will, given pk_{loss}, m, and c, simply try all possible coins to find the set of all r such that $\mathcal{E}(pk_{\mathsf{loss}}, m; r) = c$ and output a random element of that set. Nevertheless, we explicitly include the property because it is convenient in the proofs, and later we will consider variations of the definition which consider other (more efficient) opener algorithms.

We also note that the definition of lossy encryption already implies ind-cpa security. We next provide two instantiations of lossy public-key encryption, one from DDH and one from lossy trapdoor functions.

4.1 Instantiation from DDH

We now describe a lossy public-key encryption scheme based on the DDH assumption. Recall that the DDH assumption for cyclic group $\mathbb{G}$ of order prime p says that for random generator $g \in \mathbb{G}^*$ (we use $\mathbb{G}^*$ to denote the generators of $\mathbb{G}$), the tuples (g, g^a, g^b, g^{ab}) and (g, g^a, g^b, g^c) are computationally indistinguishable, where $a, b, c \leftarrow\!\!\$\ \mathbb{Z}_p$.

The scheme we describe below is originally from [36], yet some of our notation is taken from the similar dual-mode encryption scheme of [41]. The scheme has structure similar to ElGamal.

Let $\mathbb{G}$ be a prime order group of order prime p. The scheme $\mathcal{AE}_{\mathsf{ddh}} = (\mathcal{K}, \mathcal{K}_{\mathsf{loss}}, \mathcal{E}, \mathcal{D})$ is a tuple of polynomial-time algorithms defined as follows:

Algorithm $\mathcal{K}(1^\lambda)$
$g \leftarrow\!\!\$\ \mathbb{G}^*;\ x, r \leftarrow\!\!\$\ \mathbb{Z}_p$
$pk \leftarrow (g, g^r, g^x, g^{rx})$
$sk \leftarrow x$
Return (pk, sk)

Algorithm $\mathcal{E}(pk, m)$
$(g, h, g', h') \leftarrow pk$
$(u, v) \leftarrow\!\!\$\ \mathsf{Rand}(g, h, g', h')$
Return $(u, v \cdot m)$

Algorithm $\mathcal{D}(sk, c)$
$(c_0, c_1) \leftarrow c$
Return c_1 / c_0^{sk}

Algorithm $\mathcal{K}_{\mathsf{loss}}(1^\lambda)$
$g \leftarrow\!\!\$\ \mathbb{G}^*;\ r, x \neq y \leftarrow\!\!\$\ \mathbb{Z}_p$
$pk \leftarrow (g, g^r, g^x, g^{ry})$
$sk \leftarrow \perp$
Return (pk, sk)

Subroutine $\mathsf{Rand}(g, h, g', h')$
$s, t \leftarrow\!\!\$\ \mathbb{Z}_p$
$u \leftarrow g^s h^t;\ v \leftarrow (g')^s (h')^t$
Return (u, v)

We show that $\mathcal{AE}_{\mathsf{ddh}}$ satisfies the four properties of lossy encryption schemes.

1. *Correctness on real keys.* To see the correctness property is satisfied, consider a (real) public key $pk = (g, g^r, g^x, g^{rx})$ and corresponding secret key $sk = x$. Then, for some message $m \in \mathbb{G}$

$$\begin{aligned}\mathcal{D}(sk, \mathcal{E}(pk, m)) &= \mathcal{D}(sk, (g^{s+rt}, g^{xs+rxt} \cdot m)) \\ &= (g^{xs+rxt} \cdot m)/(g^{s+rt})^x \\ &= m\end{aligned}$$

2. *Indistinguishability of real keys from lossy keys.* This follows from the assumption that DDH is hard in the groups we are using, since the first output of $\mathcal{K}$ is (g, g^r, g^x, g^{rx}) and the first output of $\mathcal{K}_{\mathsf{loss}}$ is (g, g^r, g^x, g^{ry}) for $y \neq x$.
3. *Lossiness of encryption with lossy keys.* We need to show that for any lossy public key pk generated by $\mathcal{K}_{\mathsf{loss}}$, and any messages $m_0 \neq m_1 \in \mathbb{G}$, it is the case that $\mathcal{E}(pk, m_0) \equiv_s \mathcal{E}(pk, m_1)$. The results of Peikert, Vaikuntanathan, and Waters can be applied here (specifically Lemma 4 from their paper [41]). We repeat their lemma for completeness.

   Lemma 1 (Lemma 4 from [41]). *Let $\mathbb{G}$ be an arbitrary multiplicative group of prime order p. For each $x \in \mathbb{Z}_p$, define $\mathsf{DLOG}_{\mathbb{G}}(x) = \{(g, g^x) : g \in \mathbb{G}\}$. There is a probabilistic algorithm Rand that takes generators $g, h \in \mathbb{G}$ and elements $g', h' \in \mathbb{G}$, and outputs a pair $(u, v) \in \mathbb{G}^2$ such that:*
   - *If $(g, g'), (h, h') \in \mathsf{DLOG}_{\mathbb{G}}(x)$ for some x, then (u, v) is uniformly random in $\mathsf{DLOG}_{\mathbb{G}}(x)$.*
   - *If $(g, g') \in \mathsf{DLOG}_{\mathbb{G}}(x)$ and $(h, h') \in \mathsf{DLOG}_{\mathbb{G}}(y)$ for $x \neq y$, then (u, v) is uniformly random in $\mathbb{G}^2$.*

   The Rand procedure mentioned in the lemma is exactly our Rand procedure defined above. As [41] proves, this lemma shows that encryptions under a lossy key are statistically close, since such encryptions are just pairs of uniformly random group elements.
4. *Possible to claim any plaintext.* The unbounded algorithm Opener is simply the canonical opener mentioned above. Specifically, on input lossy public key $pk = (g, h, g', h')$, message $m \in \mathbb{G}$, and ciphertext $(c_1, c_2) \in \mathbb{G}^2$, it computes the set of all $s, t \in \mathbb{Z}_p$ such that $\mathsf{Rand}(g, h, g', h'; s, t)$ outputs $(c_1, c_2/m)$. It then outputs a random element of this set.

4.2 Instantiation from Lossy TDFs

Before giving our scheme we will recall a few definitions.

Definition 1 (Pairwise Independent Function Family). *A family of functions $\mathcal{H}_{n,m}$ from $\{0,1\}^n$ to $\{0,1\}^m$ is* pairwise-independent *if for any distinct $x, x' \in \{0,1\}^n$ and any $y, y \in \{0,1\}^m$,*

$$\Pr_{h \xleftarrow{\$} \mathcal{H}_{n,m}}[h(x) = y \wedge h(x') = y'] = \frac{1}{2^{2m}}.$$

For our results, we make use of *lossy trapdoor functions*, a primitive recently introduced by Peikert and Waters [40]. Informally, a lossy trapdoor function is similar to a traditional injective trapdoor function, but with the extra property that the trapdoor function is indistinguishable from another function that loses information about its input. We recall the definition from Peikert and Waters (with minor notational changes):

Definition 2 (Collection of (n,k) Lossy Trapdoor Functions). *Let λ be a security parameter, $n = n(\lambda) = \mathsf{poly}(\lambda)$, and $k = k(\lambda) \leq n$. A* collection of (n,k)-lossy trapdoor functions $\mathcal{L}_{n,k} = (S_{\mathsf{tdf}}, S_{\mathsf{loss}}, F_{\mathsf{tdf}}, F_{\mathsf{tdf}}^{-1})$ *is a tuple of algorithms with the following properties:*

1. Easy to sample, compute, and invert given a trapdoor, an injective trapdoor function. *The sampler S_{tdf}, on input 1^λ outputs (s,t), algorithm F_{tdf}, on input index s and some point $x \in \{0,1\}^n$, outputs $f_s(x)$, and algorithm F_{tdf}^{-1}, on input t and y outputs $f_s^{-1}(y)$.*
2. Easy to sample and compute lossy functions. *Algorithm S_{loss}, on input 1^λ, outputs $(s,\perp)$, and algorithm F_{tdf}, on input index s and some point $x \in \{0,1\}^n$, outputs $f_s(x)$, and the image size of f_s is at most $2^r = 2^{n-k}$.*
3. Difficult to distinguish between injective and lossy. *The function indices outputted by the sampling algorithms S_{tdf} and S_{loss} should be computationally indistinguishable. We say the advantage of distinguishing between the indices is the* ltdf*-advantage.*

We now describe an instantiation of lossy encryption based on lossy trapdoor functions.

Let λ be a security parameter and let $(S_{\mathsf{tdf}}, S_{\mathsf{loss}}, F_{\mathsf{tdf}}, F_{\mathsf{tdf}}^{-1})$ define a collection of (n,k)-lossy trapdoor functions. Also let $\mathcal{H}$ be a collection of pairwise independent hash functions from n bits to ℓ bits; the message space of the cryptosystem will then be $\{0,1\}^\ell$. The parameter ℓ should be such that $\ell \leq k - 2\log(1/\delta)$, where δ is a negligible function in the security parameter λ. The scheme $\mathcal{AE}_{\mathsf{loss}} = (\mathcal{K}, \mathcal{K}_{\mathsf{loss}}, \mathcal{E}, \mathcal{D})$ is then defined as follows:

Algorithm $\mathcal{K}(1^\lambda)$	**Algorithm** $\mathcal{E}(pk, m)$	**Algorithm** $\mathcal{D}(sk, c)$
$(s,t) \leftarrow\!\!\$\ S_{\mathsf{tdf}}(1^\lambda)$	$(s,h) \leftarrow pk$	$(t,h) \leftarrow sk$
$h \leftarrow\!\!\$\ \mathcal{H}$	$x \leftarrow\!\!\$\ \{0,1\}^n$	$(c_1,c_2) \leftarrow c$
$pk \leftarrow (s,h)$; $sk \leftarrow (t,h)$	$c_1 \leftarrow F_{\mathsf{tdf}}(s,x)$	$x \leftarrow F_{\mathsf{tdf}}^{-1}(t,c_1)$
Return (pk, sk)	$c_2 \leftarrow m \oplus h(x)$	Return $h(x) \oplus c_2$
	Return (c_1, c_2)	

The $\mathcal{K}_{\mathsf{loss}}$ algorithm is simply the same as $\mathcal{K}$, but using S_{loss} instead of S_{tdf}. (In this case, the trapdoor t will be $\perp$.)

We now show that $\mathcal{AE}_{\mathsf{loss}}$ satisfies the four properties of lossy encryption schemes.

1. *Correctness on real keys.* This follows since when $pk = (s,h)$ was generated by $\mathcal{K}$, s is such that $(s,t) \leftarrow\!\!\$\ S_{\mathsf{tdf}}(1^\lambda)$ and $h \leftarrow\!\!\$\ \mathcal{H}$ so that

$$\begin{aligned}\mathcal{D}(sk, \mathcal{E}(pk, m)) &= h(F_{\mathsf{tdf}}^{-1}(t, F_{\mathsf{tdf}}(s,x))) \oplus (m \oplus h(x)) \\ &= h(x) \oplus m \oplus h(x) \\ &= m\end{aligned}$$

2. *Indistinguishability of real keys from lossy keys.* We need to show that any efficient adversary has low $\mathsf{lossy\text{-}key}$ advantage in distinguishing between a real public key (s, h) and a lossy key (s', h'), where $(s, h) \leftarrow\!\!\$\ \mathcal{K}(1^\lambda)$ and $(s', h') \leftarrow\!\!\$\ \mathcal{K}_{\mathsf{loss}}(1^\lambda)$. Since s is the first output of S_{tdf} and s' is the first output of S_{loss}, we use the third property of lossy trapdoor functions, specifically that the function indices outputted by S_{tdf} and S_{loss} are computationally indistinguishable.
3. *Lossiness of encryption with lossy keys.* We need to show that for any lossy public key pk generated by $\mathcal{K}_{\mathsf{loss}}$, and any messages $m_0 \neq m_1 \in \{0,1\}^\ell$, it is the case that $\mathcal{E}(pk, m_0) \equiv_s \mathcal{E}(pk, m_1)$. As Peikert and Waters show in [40], this is true because of the lossiness of f_s (where s is part of pk, generated by S_{loss}). Specifically, they show that the average min-entropy $\tilde{\mathrm{H}}_\infty(x|(c_1, pk))$ of the random variable x, given $f_s(x)$ and pk is at least k, and since $\ell \leq k - 2\log(1/\delta)$, it follows that $h(x)$ will be δ-close to uniform and $m_b \oplus h(x)$ will also be δ-close to uniform for either bit b.
4. *Possible to claim any plaintext.* Again, the opener is simply the canonical opener that is guaranteed to be correct by the first three properties. Specifically, the (unbounded) algorithm Opener, on input a public key $pk = (s, h)$, message $m' \in \{0,1\}^\ell$, and ciphertext $c = (c_1, c_2) = (f_s(x), h(x) \oplus m)$ for some $x \in \{0,1\}^n$ and $m \in \{0,1\}^\ell$, must output $x' \in \{0,1\}^n$ such that $f_s(x') = c_1$ and $h(x') \oplus m' = c_2$. To do so, Opener enumerates over all $\{0,1\}^n$ and creates a set $X = \{x' \in \{0,1\}^n \ : \ f_s(x') = c_1 \wedge h(x') = m' \oplus c_2\}$ before returning a random $x \in X$.

4.3 An Extension: Efficient Opening

Recall that in the above definition of lossy encryption, the Opener algorithm could be unbounded. We will now consider a refinement of the definition that will be useful for achieving the simulation-based selective opening definition. We say that a PKE scheme $\mathcal{AE}$ is a *lossy encryption scheme with efficient opening* if it satisfies the following four properties:

1. *Correctness on real keys.* For all $(pk, sk) \leftarrow\!\!\$\ \mathcal{K}$ it must be the case that $\mathcal{D}(sk, \mathcal{E}(pk, m)) = m$.
2. *Indistinguishability of real keys from lossy keys.* No polynomial-time adversary can distinguish between the first outputs of $\mathcal{K}$ and $\mathcal{K}_{\mathsf{loss}}$.
3. *Lossiness of encryption with lossy keys.* For any $(pk, sk) \leftarrow \mathcal{K}_{\mathsf{loss}}$ and two distinct messages m_0, m_1, it must be the case that $\mathcal{E}(pk, m_0) \equiv_i \mathcal{E}(pk, m_1)$. Notice that we require ciphertexts to be identically distributed.
4. *Possible to efficiently claim any plaintext.* There exists an efficient algorithm Opener that on input lossy keys sk_{loss} and pk_{loss}, message m', and ciphertext $c = \mathcal{E}(pk_{\mathsf{loss}}, m)$, outputs an $r' \in_R \mathsf{Coins}_{\mathcal{E}}(pk_{\mathsf{loss}}, m')$ such that $\mathcal{E}(pk_{\mathsf{loss}}, m'; r') = c$. In words, the algorithm Opener is able to open ciphertexts to arbitrary plaintexts efficiently.

We emphasize that it is important for the opener algorithm to take as input the lossy secret key. This may seem strange, since in the two schemes described above the lossy secret key was simply $\perp$, but this need not be the case.

4.4 The GM Probabilistic Encryption Scheme

The Goldwasser-Micali Probabilistic encryption scheme [27] is an example of a lossy encryption scheme with efficient opening. We briefly recall the GM scheme. Let Par be an algorithm that efficiently chooses two large random primes p and q and outputs them along with their product N. Let $\mathcal{J}_p(x)$ denote the Jacobi symbol of x modulo p. We denote by QR_N the group of quadratic residues modulo N and we denote by QNR_N^{+1} the group of quadratic non-residues x such that $\mathcal{J}_N(x) = +1$. Recall that the security of the GM scheme is based on the Quadratic Residuosity Assumption, which states that it is difficult to distinguish a random element of QR_N from a random element of QNR_N^{+1}. The scheme $\mathcal{AE}_{GM} = (\mathcal{K}, \mathcal{K}_{\mathsf{loss}}, \mathcal{E}, \mathcal{D})$ is defined as follows.

Algorithm $\mathcal{K}(1^\lambda)$	**Algorithm** $\mathcal{E}(pk, m)$	**Algorithm** $\mathcal{D}(sk, \mathbf{c})$
$(N, p, q) \leftarrow\!\!\$\ \mathsf{Par}(1^\lambda)$	$(N, x) \leftarrow pk$	$(p, q) \leftarrow sk$
$x \leftarrow\!\!\$\ \mathsf{QNR}_N^{+1}$	For $i = 1$ to $\lvert m\rvert$	For $i = 1$ to $\lvert\mathbf{c}\rvert$
$pk \leftarrow (N, x)$	$r_i \leftarrow\!\!\$\ \mathbb{Z}_N^*$	If $\mathcal{J}_p(\mathbf{c}[i]) = \mathcal{J}_q(\mathbf{c}[i]) = +1$
$sk \leftarrow (p, q)$	$\mathbf{c}[i] \leftarrow r_i^2 \cdot x^{m_i} \bmod N$	$m_i \leftarrow 0$
Return (pk, sk)	Return $\mathbf{c}$	Else $m_i \leftarrow 1$
		Return m

The algorithm $\mathcal{K}_{\mathsf{loss}}$ is the same as $\mathcal{K}$ except that x is chosen at random from QR_N instead of QNR_N^{+1}; in the lossy case the secret key is still the factorization of N.

It is easy to see that the scheme $\mathcal{AE}_{GM}$ meets the first three properties of lossy PKE schemes with efficient opening: the correctness of the scheme under real keys was shown in [27], the indistinguishability of real keys from lossy keys follows directly from the Quadratic Residuosity Assumption, and encryptions under lossy keys are lossy since in that case all ciphertexts are just sequences of random quadratic residues. We claim that $\mathcal{AE}_{GM}$ is also efficiently openable. To see this consider the (efficient) algorithm Opener that takes as input secret key $sk = (p, q)$, public key $pk = (N, x)$, plaintext m, and encryption $\mathbf{c}$. For simplicity, say m has length n bits. For each $i \in [n]$, Opener uses p and q to efficiently compute the four square roots of $\mathbf{c}[i]/x^{m_i}$ and lets $\mathbf{r}[i]$ be a randomly chosen one of the four. The output of Opener is the sequence $\mathbf{r}$, which is just a sequence of random elements in $\mathbb{Z}_N^*$.

5 SOA-Security from Lossy Encryption

We now state our main results for encryption: any lossy public-key encryption scheme is ind-so-enc-secure, and any lossy public-key encryption scheme with efficient opening is sem-so-enc-secure.

Theorem 1 (Lossy Encryption implies IND-SO-ENC security). *Let λ be a security parameter, $\mathcal{AE} = (\mathcal{K}, \mathcal{K}_{lossy}, \mathcal{E}, \mathcal{D})$ be any* lossy public-key encryption scheme*, $\mathcal{M}$ any efficiently samplable distribution that supports efficient re-sampling, and A be any polynomial-time adversary corrupting $t = t(\lambda)$ parties.*

Then, there exists an unbounded lossy-ind *adversary* C *and an efficient* lossy-key *adversary* B *such that*

$$\mathbf{Adv}^{\textsf{ind-so-enc}}_{A,\mathcal{AE},\mathcal{M},n,t}(\lambda) \leq 2n \cdot \mathbf{Adv}^{\textsf{lossy-ind}}_{C,\mathcal{AE}}(\lambda) + 2 \cdot \mathbf{Adv}^{\textsf{lossy-key}}_{B,\mathcal{AE}}(\lambda).$$

Proof. We will prove the theorem using a sequence of game transitions. We start with a game that is simply the ind-so-enc experiment run with A, and end with a game in which A has no advantage, showing that each subsequent game is either computationally or statistically indistinguishable from the previous game. Now, we know that

$$\mathbf{Adv}^{\textsf{ind-so-enc}}_{A,\mathcal{AE},\mathcal{M},n,t}(\lambda) = 2\Pr[\mathbf{Exp}^{\textsf{ind-so-enc}}_{A,\mathcal{AE},\mathcal{M},n,t}(\lambda)] - 1$$

by the definition of ind-so-enc-security (see Section 3.2). We will now explain the game transitions.

- G_0: The same as the ind-so-enc experiment.
- G_1: The only change is that the A_1 is given a lossy public key and lossy ciphertexts.
- H_0: Instead of opening the ciphertexts corresponding to index I (provided by A_1) by revealing the actual coins used to generate the ciphertexts, H_0 runs the Opener algorithm on the actual messages and ciphertexts and gives A_2 the coins outputted. By the definition of the Opener algorithm (see Section 4), the coins will be correctly distributed and consistent with the ciphertexts.
- H_j: We generalize H_0 with a sequence of hybrid games. In the jth hybrid game, the first j ciphertexts given to A_1 are encryptions of dummy messages instead of the first j messages outputted by $\mathcal{M}$. Yet, the game still opens the ciphertexts for A_2 *to the actual messages produced by* $\mathcal{M}$ using the Opener algorithm.
- H_n: In the last hybrid game, A_1 is given encryptions of only the dummy message, yet A_2 receives openings of the ciphertexts to the actual messages generated by $\mathcal{M}$.

We first claim that there is an efficient adversary B such that

$$\Pr[\mathsf{G}_0] - \Pr[\mathsf{G}_1] = \mathbf{Adv}^{\textsf{lossy-key}}_{B,\mathcal{AE}}(\lambda). \tag{1}$$

To see this consider a B that is given a challenge public key pk^* and must decide whether or not it is lossy. The adversary uses the ind-so-enc-adversary A and executes exactly the same as G_0 and G_1, giving the adversary the challenge key pk^* and ciphertexts generated using pk^*. It is important for the conditional resamplability of $\mathcal{M}$ to be efficient in order for adversary B to be efficient.

Next, we claim that

$$\Pr[\mathsf{G}_1] = \Pr[\mathsf{H}_0]. \tag{2}$$

Recall that H_0 opens ciphertexts $\mathbf{c}[i] = \mathcal{E}(pk, \mathbf{m}_0[i])$ by using the Opener procedure. The key point is that in H_0, $\mathbf{c}[i]$ is still opened to $\mathbf{m}_0[i]$. This ensures us that Opener will always succeed in finding coins that open the ciphertext

correctly, and ensures us that the output of Opener is identically distributed to the actual coins used to encrypt $\mathbf{m}$. Thus, the claim follows.

We can now use a standard hybrid arguments to claim there is an *unbounded* adversary C such that

$$\Pr[\mathsf{H}_0] - \Pr[\mathsf{H}_n] = n \cdot \mathbf{Adv}^{\mathsf{lossy\text{-}ind}}_{C,\mathcal{AE}}(\lambda). \tag{3}$$

Adversary C, on input a lossy public key pk^*, will operate the same as H_j (for some guess j) except that it will use the challenge key, and for the jth ciphertext it will use the result of issuing an IND-CPA challenge consisting of the dummy message $\mathbf{m}_{dum}$ and the real message $\mathbf{m}_0[j]$. The adversary C needs to be unbounded because it runs the (possibly inefficient) procedure Opener. With standard IND-CPA, the unbounded nature of C would be problematic. However, in the case of lossy encryption, the encryptions of two distinct lossy ciphertexts are *statistically close* instead of just computationally indistinguishable, so C will still have only negligible advantage.

Finally, we claim that

$$\Pr[\mathsf{H}_n] = 1/2, \tag{4}$$

which is true since in H_n the adversary A_1 is given encryptions of dummy messages and has no information about the messages chosen from $\mathcal{M}$. (In fact, we could modify the games again and move the choice of the messages to after receiving I from A_1.)

Combining the above equations, we see that

$$\mathbf{Adv}^{\mathsf{ind\text{-}sda}}_{A,\mathcal{AE},\mathcal{M},n,t}(\lambda) \leq 2n \cdot \mathbf{Adv}^{\mathsf{lossy\text{-}ind}}_{C,\mathcal{AE}}(\lambda) + 2 \cdot \mathbf{Adv}^{\mathsf{lossy\text{-}key}}_{B,\mathcal{AE}}(\lambda),$$

which proves the theorem. □

Theorem 2 (Lossy Encryption with Efficient Opening implies SEM-SO-ENC security). *Let λ be a security parameter, $\mathcal{AE} = (\mathcal{K}, \mathcal{K}_{lossy}, \mathcal{E}, \mathcal{D})$ be any* lossy public-key encryption scheme with efficient opening*, $\mathcal{M}$ any efficiently samplable distribution, R an efficiently computable relation, and $A = (A_1, A_2)$ be any polynomial-time adversary corrupting $t = t(\lambda)$ parties. Then, there exists an efficient simulator $S = (S_1, S_2)$ and efficient* lossy-key *adversary B such that*

$$\mathbf{Adv}^{\mathit{sem\text{-}so\text{-}enc}}_{A,S,\mathcal{AE},\mathcal{M},\mathsf{R},n,t}(\lambda) \leq \mathbf{Adv}^{\mathit{lossy\text{-}key}}_{B,\mathcal{AE}}(\lambda).$$

Proof (Sketch). The proof of Theorem 2 is very similar to the proof of Theorem 1, so we will only sketch it here. For more details see [7]. We can modify the sem-so-enc-real experiment step by step until we have a successful simulator in the sem-so-enc-ideal experiment. Consider the following sequence of games:

G_0: The sem-so-enc-real experiment.
G_1: Same as G_0 except the adversary A_1 is given a lossy public key. The games are indistinguishable by the second property of efficiently openable lossy encryption.
G_2: Instead of giving A_2 the actual randomness $\mathbf{r}[I]$, the experiment uses the efficient Opener procedure.
G_3: Adversary A_1 is given encryptions of dummy messages, but A_2 is still given openings to the actual messages in $\mathbf{m}$. To do this, the efficient Opener algorithm is applied to the dummy ciphertexts.

We can then construct a simulator $S = (S_1, S_2)$ that runs A exactly as its run in G_3. Specifically, S chooses a lossy keypair and runs A_1 with a vector of encryptions of dummy messages. When A_1 outputs a set I, S asks for the same set I and learns messages $\mathbf{m}_I$. The simulator then uses the efficient Opener algorithm to open the dummy ciphertexts to the values $\mathbf{m}_I$ and finally outputs the same w as A_2. Thus, the game G_3 is identical to the sem-so-enc-ideal experiment run with simulator S. Since all of the games are close, the theorem follows. □

6 Commitment Preliminaries and Definitions

Commitment schemes

Definition 3 (Commitment scheme). *For a pair of PPT machines* $\mathsf{Com} = (\mathsf{S}, \mathsf{R})$ *and a machine* A*, consider the following experiments:*

Experiment $\mathbf{Exp}^{\mathsf{binding}}_{\mathsf{Com},A}(\lambda)$	**Experiment** $\mathbf{Exp}^{\mathsf{hiding}\text{-}b}_{\mathsf{Com},A}(\lambda)$
run $\langle \mathsf{R}(\mathtt{recv}), A(\mathtt{com})\rangle$	$(m_0, m_1) \leftarrow\!\!\$\ A(\mathtt{choose})$
$m'_0 \leftarrow\!\!\$\ \langle \mathsf{R}(\mathtt{open}), A(\mathtt{open}, 0)\rangle$	return $\langle A(\mathtt{recv}), \mathsf{S}(\mathtt{com}, m_b)\rangle$
rewind A and R back to after step 1	
$m'_1 \leftarrow\!\!\$\ \langle \mathsf{R}(\mathtt{open}), A(\mathtt{open}, 1)\rangle$	
return 1 iff $\bot \neq m'_0 \neq m'_1 \neq \bot$	

In this, $\langle A, \mathsf{S}\rangle$ *denotes the output of* A *after interacting with* S*, and* $\langle \mathsf{R}, A\rangle$ *denotes the output of* R *after interacting with* A*. We say that* Com *is a* commitment scheme *iff the following holds:*

Syntax. *For any* $m \in \{0,1\}^\lambda$*,* $\mathsf{S}(\mathtt{com}, m)$ *first interacts with* $\mathsf{R}(\mathtt{recv})$*. We call this the* commit phase*. After that,* $\mathsf{S}(\mathtt{open})$ *interacts again with* $\mathsf{R}(\mathtt{open})$*, and* R *finally outputs a value* $m' \in \{0,1\}^\lambda \cup \{\bot\}$*. We call this the* opening phase*.*

Correctness. *We have* $m' = m$ *always and for all* m*.*

Hiding. *For a PPT machine* A*, let*

$$\mathbf{Adv}^{\mathsf{hiding}}_{\mathsf{Com},A}(\lambda) := \Pr\left[\mathbf{Exp}^{\mathsf{hiding}\text{-}0}_{\mathsf{Com},A} = 1\right](\lambda) - \Pr\left[\mathbf{Exp}^{\mathsf{hiding}\text{-}1}_{\mathsf{Com},A} = 1\right](\lambda),$$

where $\mathbf{Exp}^{\mathsf{hiding}\text{-}b}_{\mathsf{Com},A}$ *is depicted below. For* Com *to be hiding, we demand that* $\mathbf{Adv}^{\mathsf{hiding}}_{\mathsf{Com},A}$ *is negligible for all PPT* A *that satisfy* $m_0, m_1 \in \{0,1\}^\lambda$ *always.*

Binding. *For a machine A, consider the experiment $\mathbf{Exp}^{\mathsf{binding}}_{\mathsf{Com},A}$ below. For* Com *to be binding, we require that $\mathbf{Adv}^{\mathsf{binding}}_{\mathsf{Com},A}(\lambda) = \Pr\left[\mathbf{Exp}^{\mathsf{binding}}_{\mathsf{Com},A}(\lambda) = 1\right]$ is negligible for all PPT A.*

Further, we say that Com *is* perfectly binding *iff $\mathbf{Adv}^{\mathsf{binding}}_{\mathsf{Com},A} = 0$ for all A. We say that* Com *is* statistically hiding *iff $\mathbf{Adv}^{\mathsf{hiding}}_{\mathsf{Com},A}$ is negligible for all (not necessarily PPT) A.*

Definition 4 (Non-interactive commitment scheme). *A* non-interactive commitment scheme *is a commitment scheme $\mathsf{Com} = (\mathsf{S}, \mathsf{R})$ in which both commit and opening phase consist of only one message sent from S to R. We can treat a non-interactive commitment scheme as a pair of algorithms rather than machines. Namely, we write $(com, dec) \leftarrow\!\!\$\ \mathsf{S}(m)$ shorthand for the commit message com and opening message dec sent by S on input m. We also denote by $m' \leftarrow\!\!\$\ \mathsf{R}(com, dec)$ the final output of R upon receiving com in the commit phase and dec in the opening phase.*

Note that perfectly binding implies that *any* commitment can only be opened to at most one value m. Perfectly binding (non-interactive) commitment schemes can be achieved from any one-way permutation (e.g., Blum [9]). On the other hand, statistically hiding implies that for any $m_0, m_1 \in \{0,1\}^\lambda$, the statistical distance between the respective views of the receiver in the commit phase is negligible. One-way functions suffice to implement statistically hiding (interactive) commitment schemes (Haitner and Reingold [29]), but there are certain lower bounds for the communication complexity of such constructions (Wee [47], Haitner et al.[30]). However, if we assume the existence of (families of) collision-resistant hash functions, then even constant-round statistically hiding commitment schemes exist (Damgard et al. [19], Naor and Yung [37]).

Interactive argument systems and zero-knowledge. We recall some basic definitions concerning interactive argument systems, mostly following Goldreich [25].

Definition 5 (Interactive proof/argument system). *An* interactive proof system *for a language $\mathcal{L}$ with witness relation $\mathcal{R}$ is a pair of PPT machines $\mathsf{IP} = (\mathsf{P}, \mathsf{V})$ such that the following holds:*

Completeness. *For every family $(x_\lambda, w_\lambda)_{\lambda\in\mathbb{N}}$ such that $\mathcal{R}(x_\lambda, w_\lambda)$ for all λ and $|x_\lambda|$ is polynomial in λ, we have that the probability for $\mathsf{V}(x_\lambda)$ to output 1 after interacting with $\mathsf{P}(x_\lambda, w_\lambda)$ is at least $2/3$.*

Soundness. *For every machine P^* and every family $(x_\lambda, z_\lambda)_{\lambda\in\mathbb{N}}$ such that $|x_\lambda| = \lambda$ and $x_\lambda \notin \mathcal{L}$ for all λ, we have that the probability for $\mathsf{V}(x_\lambda)$ to output 1 after interacting with $P^*(x_\lambda, z_\lambda)$ is at most $1/3$.*

If the soundness condition holds for all PPT machines P^* (but not necessarily for all unbounded P^*), then IP *is an* interactive argument system. *We say that* IP *enjoys* perfect completeness *if* V *always outputs 1 in the completeness condition.*

Furthermore, IP *has* negligible soundness error *if* V *outputs 1 only with negligible probability in the soundness condition.*

Definition 6 (Zero-knowledge). *Let* $\mathsf{IP} = (\mathsf{P}, \mathsf{V})$ *be an interactive proof or argument system for language* $\mathcal{L}$ *with witness relation* $\mathcal{R}$. IP *is* zero-knowledge *if for every PPT machine* V^**, there exists a PPT machine* S^* *such that for all sequences* $(x, w) = (x_\lambda, w_\lambda)_{\lambda\in\mathbb{N}}$ *with* $\mathcal{R}(x_\lambda, w_\lambda)$ *for all* λ *and* $|x_\lambda|$ *polynomial in* λ*, for all PPT machines* D*, and all auxiliary inputs* $z^{V^*} = (z_\lambda^{V^*})_{\lambda\in\mathbb{N}} \in (\{0,1\}^*)^{\mathbb{N}}$ *and* $z^D = (z_\lambda^D)_{\lambda\in\mathbb{N}} \in (\{0,1\}^*)^{\mathbb{N}}$*, we have that*

$$\mathsf{Adv}^{\mathsf{ZK}}_{V^*,S^*,(x,w),D,z^{V^*},z^D}(\lambda) := \Pr\Big[D(x_\lambda, z_\lambda^D, \langle \mathsf{P}(x_\lambda, w_\lambda), V^*(x_\lambda, z_\lambda^{V^*})\rangle) = 1\Big] - \Pr\Big[D(x_\lambda, z_\lambda^D, S^*(x_\lambda, z_\lambda^{V^*})) = 1\Big]$$

is negligible in λ*. Here* $\langle \mathsf{P}(x_\lambda, w_\lambda), V^*(x_\lambda, z_\lambda^{V^*})\rangle$ *denotes the transcript of the interaction between the prover* P *and* V^**.*

Most known interactive proof system achieve perfect completeness. Conversely, most systems do not enjoy a negligible soundness error "by nature"; their soundness has to be amplified via repetition, e.g., via sequential or concurrent composition. Thus, it is important to consider the concurrent composition of an interactive argument system:

Definition 7 (Concurrent zero-knowledge). *Let* $\mathsf{IP} = (\mathsf{P}, \mathsf{V})$ *be an interactive proof or argument system for language* $\mathcal{L}$ *with witness relation* $\mathcal{R}$. IP *is* zero-knowledge under concurrent composition *iff for every polynomial* $n = n(\lambda)$ *and PPT machine* V^**, there exists a PPT machine* S^* *such that for all sequences* $(x, w) = (x_{i,\lambda}, w_{i,\lambda})_{\lambda\in\mathbb{N}, i\in[n]}$ *with* $\mathcal{R}(x_{i,\lambda}, w_{i,\lambda})$ *for all* i, λ *and* $|x_{i,\lambda}|$ *polynomial in* λ*, for all PPT machines* D*, and all auxiliary inputs* $z^{V^*} = (z_\lambda^{V^*})_{\lambda\in\mathbb{N}} \in (\{0,1\}^*)^{\mathbb{N}}$ *and* $z^D = (z_\lambda^D)_{\lambda\in\mathbb{N}} \in (\{0,1\}^*)^{\mathbb{N}}$*, we have that*

$$\mathsf{Adv}^{\mathsf{cZK}}_{V^*,S^*,(x,w),D,z^{V^*},z^D} := \Pr\Big[D((x_{i,\lambda})_{i\in[n]}, z_\lambda^D, \langle \mathsf{P}((x_{i,\lambda}, w_{i,\lambda})_{i\in[n]}), V^*((x_{i,\lambda})_{i\in[n]}, z_\lambda^{V^*})\rangle) = 1\Big] - \Pr\Big[D((x_{i,\lambda})_{i\in[n]}, z_\lambda^D, S^*((x_{i,\lambda})_{i\in[n]}, z_\lambda^{V^*})) = 1\Big]$$

is negligible in λ*. Here* $\langle \mathsf{P}((x_{i,\lambda}, w_{i,\lambda})_{i\in[n]}), V^*((x_{i,\lambda})_{i\in[n]}, z_\lambda^{V^*})\rangle$ *denotes the transcript of the interaction between* n *copies of the prover* P *(with the respective inputs* $(x_{i,\lambda}, w_{i,\lambda})$ *for* $i = 1, \ldots, n$*) on the one hand, and* V^* *on the other hand.*

There exist interactive proof systems (with perfect completeness and negligible soundness error) that achieve Definition 7 for arbitrary NP-languages if one-way permutations exist (e.g., Richardson and Kilian [44]; see also [33, 15, 1, 23, 3] for similar results in related settings). If we assume the existence of (families of) collision-resistant hash functions, then there even exist constant-round interactive proof systems that achieve a bounded version of Definition 7 in which

the number of concurrent instances is fixed in advance Barak [1], Barak and Goldreich [2]).[2]

Black-box reductions. Reingold et al. [43] give an excellent overview and classification of black-box reductions. We recall some of their definitions which are important for our case. A *primitive* $\mathsf{P} = (F_{\mathsf{P}}, R_{\mathsf{P}})$ is a set F_{P} of functions $f : \{0,1\}^* \to \{0,1\}^*$ along with a relation R over pairs (f, A), where $f \in F_{\mathsf{P}}$, and A is a machine. We say that f is an *implementation* of P iff $f \in F_{\mathsf{P}}$. Furthermore, f is an *efficient implementation* of P iff $f \in F_{\mathsf{P}}$ and f can be computed by a PPT machine. A machine A P*-breaks* $f \in F_{\mathsf{P}}$ iff $R_{\mathsf{P}}(f, A)$. A primitive P *exists* if there is an efficient implementation $f \in F_{\mathsf{P}}$ such that no PPT machine P-breaks f. A primitive P *exists relative to an oracle* $\mathcal{B}$ iff there exists an implementation $f \in F_{\mathsf{P}}$ which is computable by a PPT machine with access to $\mathcal{B}$, such that no PPT machine with access to $\mathcal{B}$ P-breaks f.

Definition 8 (Relativizing reduction). *There exists a* relativizing reduction *from a primitive* $\mathsf{P} = (F_{\mathsf{P}}, R_{\mathsf{P}})$ *to a primitive* $\mathsf{Q} = (F_{\mathsf{Q}}, R_{\mathsf{Q}})$ *iff for every oracle* $\mathcal{B}$*, the following holds: if* Q *exists relative to* $\mathcal{B}$*, then so does* P*.*

Definition 9 (∀∃semi-black-box reduction). *There exists a* ∀∃semi-black-box reduction *from a primitive* $\mathsf{P} = (F_{\mathsf{P}}, R_{\mathsf{P}})$ *to a primitive* $\mathsf{Q} = (F_{\mathsf{Q}}, R_{\mathsf{Q}})$ *iff for every implementation* $f \in F_{\mathsf{Q}}$*, there exists a PPT machine* G *such that* $G^f \in F_{\mathsf{P}}$*, and the following holds: if there exists a PPT machine* A *such that* A^f P*-breaks* G^f*, then there exists a PPT machine* S *such that* S^f Q*-breaks* f*.*

It can be seen that if a relativizing reduction exists, then so does a ∀∃semi-black-box reduction. The converse is true when Q "allows embedding," which essentially means that additional oracles can be embedded into Q without destroying its functionality (see Reingold et al. [43], Definition 3.4 and Theorem 3.5 and Simon [46]). Below we will prove impossibility of relativizing reductions between certain primitives, which also proves impossibility of ∀∃semi-black-box reductions, since the corresponding primitives Q allow embedding.

7 Simulation-Based Commitment Security under Selective Openings

Consider the following real security game: adversary A gets, say, n commitments, and then may ask for openings of some of them. The security notion of Dwork et al. [22] requires that for any such A, there exists a simulator S that can approximate A's output. More concretely, for any relation R, we require that $R(\mathbf{m}, out_A)$ holds about as often as $R(\mathbf{m}, out_S)$, where $\mathbf{m} = (\mathbf{m}[i])_{i\in[n]}$ are the messages in the commitments, out_A is A's output, and out_S is S's output.

[2] It is common to allow the simulator S^* to be *expected polynomial-time.* In fact, the positive results [44, 33] (but not [1]) construct an expected PPT S^*. We will neglect this issue in the following, since our results do not depend the complexity of S^* (as long as S^* is not able to break an underlying computational assumption).

Formally, we get the following definition (where henceforth, $\mathcal{I}$ will denote the set of "allowed" opening sets):

Definition 10 (SEM-SO-COM). *Assume $n = n(\lambda) > 0$ is polynomially bounded, and let $\mathcal{I} = (\mathcal{I}_n)_n$ be a family of sets such that each $\mathcal{I}_n$ is a set of subsets of $[n]$. A commitment scheme $\mathsf{Com} = (\mathsf{S}, \mathsf{R})$ is* simulatable under selective openings *(short* SEM-SO-COM secure*) iff for every PPT n-message distribution $\mathcal{M}$, every PPT relation R, and every PPT machine A (the adversary), there is a PPT machine S (the simulator), such that $\mathbf{Adv}^{\mathsf{sem\text{-}so}}_{\mathsf{Com},\mathcal{M},A,S,R}$ is negligible. Here*

$$\mathbf{Adv}^{\mathsf{sem\text{-}so}}_{\mathsf{Com},\mathcal{M},A,S,R}(\lambda) := \Pr\left[\mathbf{Exp}^{\mathsf{sem\text{-}so\text{-}real}}_{\mathsf{Com},\mathcal{M},A,R} = 1\right](\lambda) - \Pr\left[\mathbf{Exp}^{\mathsf{sem\text{-}so\text{-}ideal}}_{\mathcal{M},S,R} = 1\right](\lambda),$$

where the experiments $\mathbf{Exp}^{\mathsf{sem\text{-}so\text{-}real}}_{\mathsf{Com},\mathcal{M},A,R}$ and $\mathbf{Exp}^{\mathsf{sem\text{-}so\text{-}ideal}}_{\mathcal{M},S,R}$ are defined as follows:

Experiment $\mathbf{Exp}^{\mathsf{sem\text{-}so\text{-}real}}_{\mathsf{Com},\mathcal{M},A,R}(\lambda)$	**Experiment** $\mathbf{Exp}^{\mathsf{sem\text{-}so\text{-}ideal}}_{\mathcal{M},S,R}(\lambda)$
$\mathbf{m} = (\mathbf{m}[i])_{i\in[n]} \leftarrow\!\!\$\ \mathcal{M}$	$\mathbf{m} = (\mathbf{m}[i])_{i\in[n]} \leftarrow\!\!\$\ \mathcal{M}$
$I \leftarrow\!\!\$\ \langle A(\mathtt{recv}), (\mathsf{S}_i(\mathtt{com}, \mathbf{m}[i]))_{i\in[n]}\rangle$	$I \leftarrow\!\!\$\ S(\mathtt{choose})$
$out_A \leftarrow\!\!\$\ \langle A(\mathtt{open}), (\mathsf{S}_i(\mathtt{open}))_{i\in I}\rangle$	$out_S \leftarrow\!\!\$\ S((\mathbf{m}[i])_{i\in I})$
return $R(\mathbf{m}, out_A)$	return $R(\mathbf{m}, out_S)$

In this, we require from A that $I \in \mathcal{I}_\lambda$,[3] *and we denote by $\langle A, (\mathsf{S}_i)_i\rangle$ the output of A after interacting concurrently with instances S_i of S.*

Discussion of the definitional choices. While Definition 10 essentially is the selective decommitment definition Dwork et al. [22], Definition 7.1, there are a number of definitional choices we would like to highlight (the following discussion applies equally to the upcoming Definition 13):

- Unlike [22, Definition 7.1], neither adversary A nor relation R get an auxiliary input. Such an auxiliary input is common in cryptographic definitions to ensure some form of composability.
- We do not explicitly hand the chosen set I to the relation R. Handing I to R potentially makes the definition more useful in larger contexts in which I is public.
- One could think of letting R determine the message vector $\mathbf{m}$.[4] (Equivalently, we can view $\mathcal{M}$ as part of R and let $\mathcal{M}$ forward its random coins—or a short seed—to R in a message part $\mathbf{m}[i]$ which is guaranteed not to be opened, e.g., when $i \notin I$ for all $I \in \mathcal{I}_n$.)
- The order of quantifiers $(\forall\mathcal{M}, R, A\exists S)$ is the weakest one possible. In particular, we do not mandate that S is constructed from A in a black-box way.

[3] that is, we actually only quantify over those A for which $I \in \mathcal{I}_\lambda$.

[4] This definition is closer to a universally composable definition (cf. Canetti [11]) in the sense that R (almost) takes the role of a UC-environment: R selects all inputs and reads the outputs (in particular the output of A). However, we stress that R may not actively interfere in the commitment protocol. Note that we cannot hope for *fully* UC-secure commitments for reasons not connected to the selective decommitment problem, cf. Canetti and Fischlin [12].

In all of the cases, we chose the weaker definitional variant for simplicity, which makes our negative results only stronger. We stress, however, that our positive results (Theorem 4 and Theorem 6) hold also for all of the stronger definitional variants.

7.1 Impossibility from Black-Box Reductions

Formalization of computational assumptions. Our first negative result states that SEM-SO-COM security cannot be achieved via black-box reductions from standard assumptions. We want to consider such standard assumptions in a general way that allows to make statements even in the presence of "relativizing" oracles. Thus we make the following definition, which is a special case of the definition of a *primitive* from Reingold et al. [43] (cf. also Section 6).

Definition 11 (Property of an oracle). *Let $\mathcal{X}$ be an oracle. Then a* property *$\mathcal{P}$ of $\mathcal{X}$ is a (not necessarily PPT) machine that, after interacting with $\mathcal{X}$ and another machine A, finally outputs a bit b. For an adversary A (that may interact with $\mathcal{X}$ and $\mathcal{P}$), we define A's* advantage against *$\mathcal{P}$ as*

$$\mathbf{Adv}^{\mathsf{prop}}_{\mathcal{P},\mathcal{X},A} := \Pr[\mathcal{P} \text{ outputs } b = 1 \text{ after interacting with } A \text{ and } \mathcal{X}] - 1/2.$$

Now $\mathcal{X}$ is said to satisfy *property $\mathcal{P}$ iff for all PPT adversaries A, we have that $\mathbf{Adv}^{\mathsf{prop}}_{\mathcal{P},\mathcal{X},A}$ is negligible.*

In terms of Reingold et al. [43], the corresponding primitive is $\mathsf{P} = (F_{\mathsf{P}}, R_{\mathsf{P}})$, where $F_{\mathsf{P}} = \{\mathcal{X}\}$, and $R_{\mathsf{P}}(\mathcal{X}, A)$ iff $\mathbf{Adv}^{\mathsf{prop}}_{\mathcal{P},\mathcal{X},A}$ is non-negligible. Our definition is also similar in spirit to "hard games" as used by Dodis et al. [20], but more general.

We emphasize that $\mathcal{P}$ can *only* interact with $\mathcal{X}$ and A, but not with possible additional oracles. (See Section 9 for further discussion of properties of oracles, in particular their role in our proofs.) Intuitively, $\mathcal{P}$ acts as a challenger in the sense of a cryptographic security experiment. That is, $\mathcal{P}$ tests whether adversary A can "break" $\mathcal{X}$ in the intended way. We give an example, where "breaking" means "breaking $\mathcal{X}$'s one-way property".

Example. If $\mathcal{X}$ is a random permutation of $\{0,1\}^\lambda$, then the following $\mathcal{P}$ models $\mathcal{X}$'s one-way property: $\mathcal{P}$ acts as a challenger that challenges A to invert a randomly chosen $\mathcal{X}$-image. Concretely, $\mathcal{P}$ initially chooses a random $Y \in \{0,1\}^\lambda$ and sends Y to A. Upon receiving a guess $X \in \{0,1\}^\lambda$ from A, $\mathcal{P}$ checks if $\mathcal{X}(X) = Y$. If yes, then $\mathcal{P}$ terminates with output $b = 1$. If $\mathcal{X}(X) \neq Y$, then $\mathcal{P}$ tosses an unbiased coin $b' \in \{0,1\}$ and terminates with output $b = b'$.

We stress that we only gain generality by demanding that $\Pr[\mathcal{P}$ outputs $1]$ is close to $1/2$ (and not, say, negligible). In fact, this way indistinguishability-based games (such as, e.g., the indistinguishability of ciphertexts of an ideal encryption scheme $\mathcal{X}$) can be formalized very conveniently. On the other hand, cryptographic games like the one-way game above can be formulated in this framework as well, by letting the challenger output $b = 1$ with probability $1/2$ when A fails.

On the role of property $\mathcal{P}$. Our upcoming results state the impossibility of (black-box) security reductions, from essentially *any* computational assumption (i.e., property) $\mathcal{P}$. The obvious question is: what if the assumption already *is* an idealized commitment scheme secure under selective openings? The short answer is: "then the security proof will not be black-box." We give a detailed explanation of what is going on in Section 9.

Stateless breaking oracles. In our impossibility results, we will describe a computational world with a number of oracles. For instance, there will be a "breaking oracle" $\mathcal{B}$, such that $\mathcal{B}$ aids in breaking the SEM-SO-COM security of any given commitment scheme, and in *nothing more.* To this end, $\mathcal{B}$ takes the role of the adversary in the SEM-SO-COM experiment. Namely, $\mathcal{B}$ expects to receive a number of commitments, then chooses a subset of these commitments, and then expects openings of the commitments in this subset. This is an interactive process which would usually require $\mathcal{B}$ to hold a state across invocations. However, stateful oracles are not very useful for establishing black-box separations, so we will have to give a stateless formulation of $\mathcal{B}$. Concretely, suppose that the investigated commitment scheme is non-interactive. Then $\mathcal{B}$ answers deterministically upon queries and expects each query to be prefixed with the history of that query. For instance, $\mathcal{B}$ finally expects to receive openings $dec = (dec[i])_{i\in I}$ *along* with the corresponding previous commitments $com = (com[i])_{i\in[n]}$ and previously selected set I. If I is not the set that $\mathcal{B}$ would have selected when receiving com alone, then $\mathcal{B}$ ignores the query. This way, $\mathcal{B}$ is stateless (but randomized, similarly to a random oracle). Furthermore, for non-interactive commitment schemes, this makes sure that any machine interacting with $\mathcal{B}$ can open commitments to $\mathcal{B}$ only in one way. Hence this formalization preserves the binding property of a commitment scheme, something which we will need in our proofs.

We stress, however, that this method does not necessarily work for interactive commitment schemes. Namely, any machine interacting with such a stateless $\mathcal{B}$ can essentially "rewind" $\mathcal{B}$ during an interactive commitment phase, since $\mathcal{B}$ formalizes a next-message function. Now if the commitment scheme is still binding if the receiver of the commitment can be rewound (e.g., this holds trivially for non-interactive commitment schemes, and also for perfectly binding commitment schemes), then our formalization of $\mathcal{B}$ preserves binding, and our upcoming proof works. If, however, the commitment scheme loses its binding property if the receiver can be rewound, then the following theorem cannot be applied.

We are now ready to state our result.

Theorem 3 (Black-box impossibility of non-interactive or perfectly binding SEM-SO-COM, most general formulation). *Let $n = n(\lambda) = 2\lambda$, and let $\mathcal{I} = (\mathcal{I}_n)_n$ with $\mathcal{I}_n = \{I \subseteq [n] \mid |I| = n/2\}$ denote the set of all $n/2$-sized subsets of $[n]$.*[5] *Let $\mathcal{X}$ be an oracle that satisfies property $\mathcal{P}$. Then there is a set of oracles relative to which $\mathcal{X}$ still satisfies property $\mathcal{P}$, but there exists no*

[5] We stress that the proofs of Theorem 3 and Theorem 5 hold literally also for the "cut-and-choose" $\mathcal{I}_n = \{I \subseteq [n] \mid \forall i \in [\lambda] :$ either $2i - 1 \in I$ or $2i \in I\}$.

non-interactive or perfectly binding commitment scheme which is simulatable under selective openings.

Proof strategy. We will use a random oracle $\mathcal{RO}$ that, for any given non-interactive commitment scheme Com^*, induces a message distribution $\mathcal{M}^* = \{(\mathcal{RO}(\mathsf{Com}^*, i, X^*))_{i\in[n]}\}_{X^*\in\{0,1\}^{\lambda/3}}$. Here, $\mathcal{RO}(\mathsf{Com}^*)$ denotes the hash of the description of Com^*, and X^* is a short "seed" that ties the values $\mathcal{RO}(\mathsf{Com}^*, i, X^*)$ (with the same X^* but different i) together. Furthermore, we will specify an oracle $\mathcal{B}$ that will help to break Com^* with respect to $\mathcal{M}^*$. Concretely, $\mathcal{B}$ first expects n Com^*-commitments, and then requests openings of a random subset of them. If all openings are valid, $\mathcal{B}$ returns a value X^* consistent (according to $\mathcal{M}^*$) with all opened messages (if such an X^* exists). A suitable SEM-SO-COM adversary A can use $\mathcal{B}$ simply by relaying its challenge to obtain X^* and hence the whole message vector in its SEM-SO-COM experiment.

However, we will prove that $\mathcal{B}$ is useless to any simulator S that gets only a message subset $\mathbf{m}[I]$: if S uses $\mathcal{B}$ *before* requesting its own message subset $\mathbf{m}[I]$, then $\mathcal{B}$'s answer will not be correlated with the SEM-SO-COM challenge message vector $\mathbf{m}$. (This also holds if S first sends commitments to $\mathcal{B}$ and immediately afterwards requests $\mathbf{m}[I]$ from the SEM-SO-COM experiment; in that case, S has to break the binding property of Com^* to get an answer from $\mathcal{B}$ which is correlated with $\mathbf{m}$.) But if S uses $\mathcal{B}$ *after* obtaining $\mathbf{m}[I]$, then with very high probability, S will have open at least one commitment to $\mathcal{B}$ whose message is not contained in $\mathbf{m}[I]$. By definition of $\mathcal{M}^*$, this opening of S will not be consistent with the other values of $\mathbf{m}[I]$ (except with small probability), and $\mathcal{B}$'s answer will again not be correlated with $\mathbf{m}$.

Since S cannot efficiently extract the seed X^* from its message subset $\mathbf{m}[I]$ alone (that would require a brute-force search over exponentially many values), this shows that Com^* is not SEM-SO-COM secure. Consequently, because Com^* was arbitrary (only the message distribution $\mathcal{M}^*$ is specific to Com^*), there exist no SEM-SO-COM secure commitment schemes relative to $\mathcal{RO}$ and $\mathcal{B}$. Finally, it is easy to see that relative to $\mathcal{RO}$ and $\mathcal{B}$, primitive $\mathcal{X}$ still satisfies property $\mathcal{P}$. Concretely, observe that $\mathcal{B}$ does not break any commitment (note that $\mathcal{B}$'s answer depends only on the *opened* commitments), but only inverts a message distribution (or, rather, $\mathcal{RO}$). Hence, any adversary attacking property $\mathcal{P}$ of $\mathcal{X}$ can use efficient internal simulations of $\mathcal{RO}$ and $\mathcal{B}$ instead of the original oracles. Since $\mathcal{X}$ satisfies property $\mathcal{P}$ with respect to adversaries without (additional) oracle access, the claim follows.

The following corollary provides an instantiation of Theorem 3 for a number of standard cryptographic primitives.

Corollary 1 (Black-box impossibility of non-interactive or perfectly binding SEM-SO-COM). *Assume n and $\mathcal{I}$ as in Theorem 3. Then no non-interactive or perfectly binding commitment scheme can be proved simulatable under selective openings via a $\forall\exists$semi-black-box reduction to one or more of the following primitives: one-way functions, one-way permutations, trapdoor one-way permutations, IND-CCA secure public key encryption, homomorphic public key encryption.*

The corollary is a special case of Theorem 3. For instance, to show Corollary 1 for one-way permutations, one can use the example $\mathcal{X}$ and $\mathcal{P}$ from above: $\mathcal{X}$ is a random permutation of $\{0,1\}^\lambda$, and $\mathcal{P}$ models the one-way experiment with $\mathcal{X}$. Clearly, $\mathcal{X}$ satisfies $\mathcal{P}$, and so we can apply Corollary 1. This yields impossibility of relativizing proofs for SEM-SO-COM security from one-way permutations. We get impossibility for $\forall\exists$semi-black-box reductions since one-way permutations allow embedding, cf. Simon [46],Reingold et al. [43]. The other cases are similar. Note that while it is generally not easy to even give a candidate for a cryptographic primitive in the standard model, it is easy to construct an idealized, say, encryption scheme in oracle form.

We stress that Corollary 1 makes no assumptions about the nature of the simulation (in the sense of Definition 10). In particular, the simulator may freely use, e.g., the code of the adversary; the only restriction is black-box access to the underlying primitive. As discussed in the introduction, this is quite different from the result one gets upon combining Goldreich and Krawczyk [26] and Dwork et al. [22]: essentially, combining [26, 22] shows impossibility of constructing S in a black-box way from A (i.e., such that S only gets black-box access to A's next-message function).

Generalizations. First, Corollary 1 constitutes merely an example instantiation of the much more general Theorem 3. Second, the proof also holds for a relaxation of SEM-SO-COM security considered by Dwork et al. [22], Definition 7.3, where adversary and simulator approximate a function of the message vector.

7.2 Possibility Using Non-black-box Techniques

Non-black-box techniques vs. interaction. Theorem 3 shows that SEM-SO-COM security cannot be achieved unless one uses non-black-box techniques or interaction. In this section, we will investigate the power of non-black-box techniques to achieve SEM-SO-COM security. As it turns out, for our purposes a concurrently composable zero-knowledge argument system is a suitable non-black-box tool.[6] We stress that the use of this zero-knowledge argument makes our scheme necessarily interactive, and so actually circumvents Theorem 3 in *two* ways: by non-black-box techniques *and* by interaction. However, from a conceptual point of view, our scheme is "non-interactive up to the zero-knowledge argument." In particular, our proof does not use the fact that the zero-knowledge argument is interactive. (That is, if we used a concurrently composable non-interactive zero-knowledge argument in, say, the common reference string model, our proof would still work.)

The scheme. For our non-black-box scheme, we need an interactive argument system IP with perfect completeness and negligible soundness error, such that IP is zero-knowledge under concurrent composition. We also need a perfectly binding non-interactive commitment scheme $\mathsf{Com}^{\mathsf{b}}$. Both these ingredients can be

[6] We require concurrent composability since the SEM-SO-COM definition considers multiple, concurrent sessions of the commitment scheme.

constructed from one-way permutations. To ease presentation, we only describe a *bit* commitment scheme, which is easily extended (along with the proof) to the multi-bit case. In a nutshell, the sender S^{ZK} commits twice (using $\mathsf{Com}^{\mathsf{b}}$) to the the same bit and proves in zero-knowledge (using IP) that the committed bits are the same.[7] In the opening phase, the sender opens one (randomly selected) commitment. Note that this overall commitment scheme is binding, since IP ensures that both commitments contain the same bits, and the underlying commitment $\mathsf{Com}^{\mathsf{b}}$ is binding. For a SEM-SO-COM simulation, we generate inconsistent overall commitments which can later be opened arbitrarily by choosing which individual $\mathsf{Com}^{\mathsf{b}}$-commitment is opened. We can use the simulator of IP to generate fake consistency proofs for these inconsistent commitments. (Since we consider many concurrent commitment instances in our SEM-SO-COM experiment, we require concurrent composability from IP for that.)

Scheme 12 (Non-black-box commitment scheme ZKCom). Let $\mathsf{Com}^{\mathsf{b}} = (\mathsf{S}^{\mathsf{b}}, \mathsf{R}^{\mathsf{b}})$ be a perfectly binding non-interactive commitment scheme. Let $\mathsf{IP} = (\mathsf{P}, \mathsf{V})$ be an interactive argument system for NP which enjoys perfect completeness, has negligible soundness error, and which is zero-knowledge under concurrent composition. Let $\mathsf{ZKCom} = (\mathsf{S}^{\mathsf{ZK}}, \mathsf{R}^{\mathsf{ZK}})$ for the following S^{ZK} and R^{ZK}:

- Commitment to bit b:
  1. S^{ZK} prepares $(com^j, dec^j) \leftarrow\!\!\$\ \mathsf{S}^{\mathsf{b}}(b)$ for $j \in \{0,1\}$ and sends (com^0, com^1) to R^{ZK}.
  2. S^{ZK} uses IP to prove to R^{ZK} that com^0 and com^1 commit to the same bit.[8]
- Opening:
  1. S^{ZK} uniformly chooses $j \in \{0,1\}$ and sends (j, dec^j) to R^{ZK}.

The security of ZKCom. It is straightforward to prove that ZKCom is a hiding and binding commitment scheme. (We stress, however, that $\mathsf{Com}^{\mathsf{b}}$'s *perfect* binding property is needed to prove that ZKCom is binding; otherwise, the zero-knowledge argument may become meaningless.) More interestingly, we can also show that ZKCom is SEM-SO-COM secure:

Theorem 4 (Non-black-box possibility of SEM-SO-COM). *Fix n and $\mathcal{I}$ as in Definition 10. Then* ZKCom *is simulatable under selective openings in the sense of Definition 10.*

[7] We note that a FOCS referee, reviewing an earlier version of this paper without ZKCom, also suggested to employ zero-knowledge to prove consistency of a given commitment. This suggestion was independent of the eprint version of this paper which at that time already contained our scheme ZKCom. A Eurocrypt referee, reviewing a version of the paper with ZKCom, remarked that alternative constructions of a SEM-SO-COM secure commitment scheme are possible. A more generic construction could be along the lines of "commit using a perfectly binding commitment, then prove consistency of commitment or opening using concurrent zero-knowledge."

[8] Formally, the corresponding language $\mathcal{L}$ for IP consists of statements $x = (com^0, com^1)$ and witnesses $w = (dec^0, dec^1)$ such that $\mathcal{R}(x,w)$ iff $\mathsf{R}^{\mathsf{b}}(com^0, dec^0) = \mathsf{R}^{\mathsf{b}}(com^1, dec^1) \in \{0,1\}$.

Proof outline. We start with the real SEM-SO-COM experiment with an arbitrary adversary A. As a first step, we substitute the proofs generated during the commitments by *simulated proofs*. Concretely, we hand to A proofs for the consistency of the commitments that are generated by a suitable simulator S^*. By the concurrent zero-knowledge property of IP, such an S^* exists and yields indistinguishable experiment outputs. Note that S^* does not need witnesses to generate valid-looking proofs, but instead uses (possibly rewinding or even non-black-box) access to A. Hence, we can substitute all ZKCom-commitments with inconsistent commitments of the form (com^0, com^1), where com^0 and com^1 are Com^b-commitments to *different* bits. Such a ZKCom-commitment can later be opened arbitrarily. By the computational hiding property of Com^b (and since we do not need witnesses to generate consistency proofs anymore), this step does not change the output distribution of the experiment significantly. But note that now, the initial generation of the commitments does not need knowledge of the actual messages. In fact, only the messages $\mathbf{m}[I]$ of the actually opened commitments need to be known at opening time. Hence, at this point, the modified experiment is a valid simulator in the sense of the ideal SEM-SO-COM experiment. Since the experiment output has only been changed negligibly by our modifications, we have thus constructed a successful simulator in the sense of Definition 10.

Where is the non-black-box component? Interestingly, the used argument system IP itself can well be black-box zero-knowledge (where black-box zero-knowledge means that the simulator S^* from Definition 7 has only black-box access to the next-message function of V^*). The essential fact that allows us to circumvent our negative result Theorem 3 is the way we employ IP. Namely, ZKCom uses IP to prove a statement about two given commitments (com^0, com^1). This proof (or, rather, argument) uses an explicit and non-black-box description of the employed commitment scheme Com^b. It is this argument that cannot even be expressed when Com^b makes use of, say, a one-way function given in oracle form.

The role of the commitment randomness. Observe that the opening of a ZKCom-commitment does not release all randomness used for constructing the commitment. In fact, it is easy to see that our proof would not hold if S^{ZK} opened *both* commitments com^0 and com^1 in the opening phase. Hence, ZKCom is not suitable for settings in which an opening corresponds to a corruption of a party (e.g., in a multi-party computation setting), and when one cannot assume no trusted erasures.

Generalizations. First, ZKCom can be straightforwardly extended to a multi-bit commitment scheme, e.g., by running several sessions of ZKCom in parallel. Second, ZKCom is SEM-SO-COM secure also against adversaries with auxiliary input z: our proof holds literally, where of course we also require security of Com^b against non-uniform adversaries.

8 Indistinguishability-Based Commitment Security under Selective Openings

Motivated by the impossibility result from the previous section, we now relax Definition 10 as follows:

Definition 13 (IND-SO-COM). *Let $n = n(\lambda) > 0$ be polynomially bounded, and let $\mathcal{I} = (\mathcal{I}_n)_n$ be a family of sets such that each $\mathcal{I}_n$ is a set of subsets of $[n]$. A commitment scheme $\mathsf{Com} = (\mathsf{S}, \mathsf{R})$ is* indistinguishable under selective openings *(short* IND-SO-COM secure*) iff for every PPT n-message distribution $\mathcal{M}$, and every PPT adversary A, we have that $\mathbf{Adv}^{\mathsf{ind\text{-}so}}_{\mathsf{Com},\mathcal{M},A}$ is negligible. Here*

$$\mathbf{Adv}^{\mathsf{ind\text{-}so}}_{\mathsf{Com},\mathcal{M},A}(\lambda) := \Pr\left[\mathbf{Exp}^{\mathsf{ind\text{-}so\text{-}real}}_{\mathsf{Com},\mathcal{M},A} = 1\right](\lambda) - \Pr\left[\mathbf{Exp}^{\mathsf{ind\text{-}so\text{-}ideal}}_{\mathsf{Com},\mathcal{M},A} = 1\right](\lambda),$$

where the experiments $\mathbf{Exp}^{\mathsf{ind\text{-}so\text{-}real}}_{\mathsf{Com},\mathcal{M},A}$ and $\mathbf{Exp}^{\mathsf{ind\text{-}so\text{-}ideal}}_{\mathsf{Com},\mathcal{M},A}$ are defined as follows:

Experiment $\mathbf{Exp}^{\mathsf{ind\text{-}so\text{-}real}}_{\mathsf{Com},\mathcal{M},A}(\lambda)$	**Experiment** $\mathbf{Exp}^{\mathsf{ind\text{-}so\text{-}ideal}}_{\mathsf{Com},\mathcal{M},A}(\lambda)$
$\mathbf{m} = (\mathbf{m}[i])_{i\in[n]} \leftarrow\!\!\$\ \mathcal{M}$	$\mathbf{m} = (\mathbf{m}[i])_{i\in[n]} \leftarrow\!\!\$\ \mathcal{M}$
$I \leftarrow\!\!\$\ \langle A(\mathtt{recv}), (\mathsf{S}_i(\mathtt{com}, \mathbf{m}[i]))_{i\in[n]}\rangle$	$I \leftarrow\!\!\$\ \langle A(\mathtt{recv}), (\mathsf{S}_i(\mathtt{com}, \mathbf{m}[i]))_{i\in[n]}\rangle$
$out_A \leftarrow\!\!\$\ \langle A(\mathtt{open}), (\mathsf{S}_i(\mathtt{open}))_{i\in I}\rangle$	$out_A \leftarrow\!\!\$\ \langle A(\mathtt{open}), (\mathsf{S}_i(\mathtt{open}))_{i\in I}\rangle$
	$\mathbf{m}' \leftarrow\!\!\$\ \mathcal{M} \mid \mathbf{m}[I]$
return $A(\mathtt{guess}, \mathbf{m})$	return $A(\mathtt{guess}, \mathbf{m}')$

Again, we require from A that $I \in \mathcal{I}_\lambda$, and we denote by $\langle A, (\mathsf{S}_i)_i\rangle$ the output of A after interacting concurrently with instances S_i of S. Furthermore, $\mathcal{M} \mid \mathbf{m}[I]$ denotes the message distribution $\mathcal{M}$ conditioned on the values of $\mathbf{m}[I]$.

On the conditioned distribution $\mathcal{M} \mid \mathbf{m}[I]$. We stress that, depending on $\mathcal{M}$, it may be computationally hard to sample $\mathbf{m}' \leftarrow\!\!\$\ \mathcal{M} \mid \mathbf{m}[I]$, even if (the unconditioned) $\mathcal{M}$ is PPT. This might seem strange at first and inconvenient when *applying* the definition in some larger reduction proof. However, there simply seems to be no other way to capture indistinguishability, since the set of opened commitments depends on the commitments themselves. In particular, in general we cannot predict which commitments the adversary wants opened, and then, say, substitute the not-to-be-opened commitments with random commitments. What we chose to do instead is to give the adversary either the full message vector, or an independent message vector which "could be" the full message vector, given the opened commitments. We believe that this is the canonical way to capture secrecy of the unopened commitments under selective openings.

The relation between SEM-SO-COM and IND-SO-COM security. Unfortunately, we (currently) cannot prove that SEM-SO-COM security implies IND-SO-COM security (although this seems plausible, since usually simulation-based definitions imply their indistinguishability-based counterparts). Technically, the reason why we are unable to prove an implication is the conditioned distribution $\mathcal{M} \mid \mathbf{m}[I]$ in the ideal IND-SO-COM experiment, which cannot be sampled from during an (efficient) reduction.

A relaxation. Alternatively, we could let the adversary predict a predicate π of the whole message vector, and consider him successful if $\Pr[b = \pi(\mathbf{m})]$ and $\Pr[b = \pi(\mathbf{m}')]$ for the alternative message vector $\mathbf{m}' \leftarrow\!\!\$\ \mathcal{M} \mid \mathbf{m}[I]$ differ non-negligibly. We stress that our upcoming negative result also applies to this relaxed notion.

8.1 Impossibility from Black-Box Reductions

Theorem 5 (Black-box impossibility of perfectly binding IND-SO-COM, most general formulation). *Let $n = n(\lambda) = 2\lambda$, and let $\mathcal{I} = (\mathcal{I}_n)_n$ with $\mathcal{I}_n = \{I \subseteq [n] \mid |I| = n/2\}$ denote the set of all $n/2$-sized subsets of $[n]$. Let $\mathcal{X}$ be an oracle that satisfies a property $\mathcal{P}$ even in presence of an EXPSPACE-oracle. We also assume that $\mathcal{X}$ is computable in EXPSPACE.*[9] *Then, there exists a set of oracles relative to which $\mathcal{X}$ still satisfies $\mathcal{P}$, but no perfectly binding commitment scheme is indistinguishable under selective openings.*

Proof outline. Similarly to Theorem 3, we specify an oracle $\mathcal{RO}$ which induces a message distribution $\mathcal{M}^*$. This time, however, $\mathcal{RO}$ maps $\mathbb{E}^{n/2+1}$-elements to message vectors in $\mathbb{E}^n$, where $\mathbb{E} = \{0,1\}^\lambda$ is the domain of each individual message. Hence, $n/2$ messages usually do not fix the whole message vector, but more messages do. Now fix any perfectly binding commitment scheme Com^*. We define a breaking oracle $\mathcal{B}$ that, like the $\mathcal{B}$ from Theorem 3, asks for n Com^*-commitments and subsequent openings of a random subset $I \in \mathcal{I}_n$ of these commitments. If all openings are valid, $\mathcal{B}$ extracts the *whole* message vector in the commitments (note that this is possible since Com^* is perfectly binding), and returns a "close" (with respect to Hamming distance) element in the message distribution $\mathcal{M}^*$ if there is a sufficiently close one.

It is easy to see that an adversary can use $\mathcal{B}$ to obtain the whole message vector $\mathbf{m}$ in the real IND-SO-COM experiment. But a message vector freshly sampled from $\mathcal{M}^*$, conditioned on the opened messages $\mathbf{m}[I]$, will most likely be different from $\mathbf{m}$. Hence, our adversary easily distinguishes the real from the ideal IND-SO-COM experiment.

The main part of the proof shows that oracle $\mathcal{B}$ is useless to an adversary attacking $\mathcal{X}$'s property $\mathcal{P}$. Assume first that the commitment scheme Com with respect to which an adversary A on $\mathcal{X}$ queries $\mathcal{B}$ is perfectly binding. In that case, a somewhat technical but straightforward combinatorial argument shows that A's successfully opened messages $\mathbf{m}[I]$, *together with A's queries to $\mathcal{RO}$,* determine $\mathcal{B}$'s answer (except with small probability). Hence A can use internal simulations of $\mathcal{B}$ and $\mathcal{RO}$ instead of the original oracles, and hence property $\mathcal{P}$ of $\mathcal{X}$ is not damaged by the presence of $\mathcal{B}$. To ensure that $\mathcal{B}$ is only useful for perfectly binding commitment schemes Com, we let $\mathcal{B}$ *test* whether Com is perfectly binding. Since we demand that Com is *perfectly* binding, this test is independent of the random coins used by $\mathcal{X}$. Indeed, $\mathcal{B}$ needs to check that for all syntactically possible commitments and decommitments, and *all* possible

[9] Examples of such $\mathcal{X}$ are random oracles or ideal ciphers. It will become clearer how we use the EXPSPACE requirement in the proof.

random coins used by $\mathcal{X}$, the opened message is unique. Hence, by assumption about $\mathcal{X}$, this test can also be performed by A using an EXPSPACE-oracle, and the above proof idea applies.

On the requirement on $\mathcal{X}$. We stress that the requirement in Theorem 5 on $\mathcal{X}$ is a rather mild one. For instance, random oracles are one-way even against computationally unbounded adversaries, as long as the adversary makes only a polynomial number of oracle queries. Hence, an EXPSPACE-oracle (which itself does not perform oracle queries) is not helpful in breaking a random oracle. So similarly to Corollary 1, we get for concrete choices of $\mathcal{X}$ and $\mathcal{P}$:

Corollary 2 (Black-box impossibility of perfectly binding IND-SO-COM). *Let n and $\mathcal{I}$ as in Theorem 5. Then no perfectly binding commitment scheme can be proved indistinguishable under selective openings via a $\forall\exists$semi-black-box reduction to one or more of the following primitives: one-way functions, one-way permutations, trapdoor one-way permutations, IND-CCA secure public key encryption, homomorphic public key encryption.*

Generalizations. Again, Corollary 2 constitutes merely an example instantiation of the much more general Theorem 5. We stress, however, that the proof for Theorem 5 does *not* apply to "almost-perfectly binding" commitment schemes such as the one from Naor [35]. (For instance, for such schemes, $\mathcal{B}$'s check that the supplied commitment scheme is binding might tell something about $\mathcal{X}$.)

8.2 Statistically Hiding Schemes Are Secure

Fortunately, things look different for statistically hiding commitment schemes:

Theorem 6 (Statistically hiding schemes are IND-SO-COM secure). *Fix arbitrary n and $\mathcal{I}$ as in Definition 13, and let $\mathsf{Com} = (\mathsf{S}, \mathsf{R})$ be a statistically hiding commitment scheme. Then Com is indistinguishable under selective openings in the sense of Definition 13.*

Proof outline. Intuitively, the claim holds since an adversary A's views in the real, resp. ideal IND-SO-COM experiment are statistically close (and hence so must be A's outputs). However, the fact that A's views are indeed statistically close is less obvious than it may seem at first glance. Our proof proceeds in games and starts with the real IND-SO-COM experiment with A. As a first modification, we change the opening phase of the experiment, so that the opening of each selected commitment is produced solely from the commitment itself and the "target message" $\mathbf{m}[i]$ to which it should be opened (but not from opening information previously generated alongside the commitment). Note that this change is merely conceptual and does not alter A's view at all. This makes the opening phase inefficient, but since we are dealing with statistically hiding commitment schemes, we need not worry about that. Indeed, by the statistical hiding property, we can now substitute all commitments (in a hybrid argument) with commitments to a fixed value (say, 0^λ) without affecting the experiment

output. We can reduce this step to the hiding property of the commitment scheme since the experiment only needs commitments as input, and produces all openings on its own. At this point, all commitments that A gets are independent of $\mathbf{m}$, and so the whole view of A is independent of the unopened values $\mathbf{m}[[n] \setminus I]$. Hence A's output is (almost) independent of $\mathbf{m}[[n] \setminus I]$ in the real IND-SO-COM experiment and, with similar reasoning, also in the ideal IND-SO-COM experiment. This shows the claim.

9 On the Role of Property $\mathcal{P}$

The intuitive contradiction. The formulations of Theorem 3 and Theorem 5 seem intuitively much too general: essentially they claim impossibility of black-box proofs from *any* computational assumption which is formulated as a property $\mathcal{P}$ of an oracle $\mathcal{X}$. Why can't we choose $\mathcal{X}$ to be an ideally secure commitment scheme, and $\mathcal{P}$ a property that models precisely what we want to achieve, e.g., Definition 13 (i.e., IND-SO-COM security)? After all, Definition 13 can be rephrased as a property $\mathcal{P}$ by letting A choose a message distribution $\mathcal{M}$ and send this distribution (as a description of a PPT algorithm $\mathcal{M}$) to $\mathcal{P}$. Then, $\mathcal{P}$ could perform the $\mathbf{Exp}^{\mathsf{ind\text{-}so\text{-}real}}_{\mathsf{Com},\mathcal{M},A}$ or the $\mathbf{Exp}^{\mathsf{ind\text{-}so\text{-}ideal}}_{\mathsf{Com},\mathcal{M},A}$ experiment with A, depending on an internal coin toss (the output of $\mathcal{P}$ will then depend on A's output and on that coin toss). This $\mathcal{P}$ models Definition 13, in the sense that

$$\mathbf{Adv}^{\mathsf{ind\text{-}so}}_{\mathsf{Com},\mathcal{M},A} = 2\mathbf{Adv}^{\mathsf{prop}}_{\mathcal{P},\mathcal{X},A}.$$

Also, using a truly random permutation as a basis, it is natural to assume that we can construct an *ideal* (i.e., as an oracle) perfectly binding commitment scheme $\mathcal{X}$ that satisfies $\mathcal{P}$. (Note that although $\mathcal{X}$ is perfectly binding, A's view may still be almost statistically independent of the unopened messages, since the scheme $\mathcal{X}$ is given in oracle form.)

Hence, if the assumption essentially *is* already IND-SO-COM security, we can certainly achieve IND-SO-COM security (in particular, using a trivial reduction), and this seems to contradict Theorem 5. So where is the problem?

Resolving the situation. The problem in the above argument is that $\mathcal{P}$-security (our assumption) implies IND-SO-COM security (our goal) in a fundamentally non-black-box way. Namely, the proof converts an IND-SO-COM adversary A and a message distribution $\mathcal{M}$ into a $\mathcal{P}$-adversary A' that sends a description of $\mathcal{M}$ to $\mathcal{P}$. This very step makes use of an *explicit representation* of the message distribution $\mathcal{M}$, and this is what makes the whole proof non-black-box. In other words, this way of achieving IND-SO-COM security cannot be black-box, and there is no contradiction to our results.

Viewed from a different angle, the essence of our impossibility proofs is: build a very specific message distribution, based on oracles ($\mathcal{RO}$, resp. $\mathcal{C}$), such that another "breaking oracle" $\mathcal{B}$ "breaks" this message distribution if and only if the adversary can prove that he can open commitments. This step relies on the fact that we can specify message distributions which depend on oracles. Relative to such oracles, property $\mathcal{P}$ still holds (as we prove), but may not reflect

IND-SO-COM security anymore. Namely, since $\mathcal{P}$ itself cannot access additional oracles[10], $\mathcal{P}$ is also not able to sample a message space that depends on additional (i.e., on top of $\mathcal{X}$) oracles. So in our reduction, although A itself can, both in the IND-SO-COM experiment and when interacting with $\mathcal{P}$, access all oracles, it will not be able to communicate a message distribution $\mathcal{M}$ that depends on additional oracles (on top of $\mathcal{X}$) to $\mathcal{P}$. On the other hand, any PPT algorithm $\mathcal{M}$, as formalized in Definition 13, *can* access all available oracles.

So for the above modeling of IND-SO-COM as a property $\mathcal{P}$ in the sense of Definition 11, our impossibility results still hold, but become meaningless (since basically using property $\mathcal{P}$ makes the proof non-black-box). In a certain sense, this comes from the fact that the modeling of IND-SO-COM as a property $\mathcal{P}$ is inherently non-black-box. A similar argument holds for the message distribution in the SEM-SO-COM experiment; there, however, we face the additional problem of modeling the existence of a simulator in a property.

What computational assumptions can be formalized as properties in a "black-box" way? Fortunately, most standard computational assumptions can be modeled in a black-box way as a property $\mathcal{P}$. Besides the mentioned one-way property (and its variants), in particular, e.g., the IND-CCA security game for encryption schemes can be modeled. Observe that in this game, we can let the IND-CCA adversary himself sample challenge messages m_0, m_1 for the IND-CCA experiment from his favorite distribution; no PPT algorithm has to be transported to the security game. In fact, the only properties which do not allow for black-box proofs are those that involve an explicit transmission of code (i.e., a description of a circuit or a Turing machine). In that sense, the formulation of Theorem 3 and Theorem 5 is very general and useful.

(Non-)programmable random oracles. We stress that the black-box requirement for random oracles (when used in the role of $\mathcal{X}$) corresponds to "non-programmable random oracles" (as used by, e.g., Bellare and Rogaway [5]) as opposed to "programmable random oracles" (as used by, e.g., Nielsen [38]). Roughly, a proof in the programmable random oracle model translates an attack on a cryptographic scheme into an attack on a *simulated* random oracle (that is, an oracle completely under control of simulator). Naturally, such a reduction is not black-box. And indeed, with programmable random oracles, even non-interactive SEM-SO-COM secure commitment schemes can be built relatively painlessly. As an example, [38] proves a simple encryption scheme (which can be interpreted as a non-interactive commitment scheme) secure under selective openings.

Acknowledgements

Bellare and Yilek thank Saurabh Panjwani for participating in early stages of this work, which involved the development of the indistinguishability-based definition

[10] by Definition 11, $\mathcal{P}$ must be specified independently of additional oracles; if we did allow $\mathcal{P}$ to access additional oracles, this would break our impossibility proofs

IND-SO-ENC. Hofheinz would like to thank Enav Weinreb, Marc Stevens, Serge Fehr, Krzysztof Pietrzak, and Ivan Damgård for many insightful discussions.

Mihir Bellare is supported by NSF grants CNS–0524765 and CNS–0627779 and a gift from Intel Corporation. Dennis Hofheinz is supported by NWO. Scott Yilek is supported by NSF grants CNS–0430595 and CNS–0831536.

References

[1] Barak, B.: How to go beyond the black-box simulation barrier. In: 42nd Annual Symposium on Foundations of Computer Science, Proceedings of FOCS 2001, pp. 106–115. IEEE Computer Society, Los Alamitos (2001)

[2] Barak, B., Goldreich, O.: Universal arguments and their applications. In: 17th Annual IEEE Conference on Computational Complexity, Proceedings of CoCo 2002, pp. 194–203. IEEE Computer Society, Los Alamitos (2002)

[3] Barak, B., Prabhakaran, M., Sahai, A.: Concurrent non-malleable zero-knowledge. In: 47th Annual Symposium on Foundations of Computer Science, Proceedings of FOCS 2006, pp. 345–354. IEEE Computer Society, Los Alamitos (2006)

[4] Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: 1st ACM Conference on Computer and Communications Security, Proceedings of CCS 1993, pp. 62–73. ACM Press, New York (1993)

[5] Bellare, M., Rogaway, P.: Optimal asymmetric encryption—how to encrypt with RSA. In: De Santis, A. (ed.) EUROCRYPT 1994. LNCS, vol. 950, pp. 92–111. Springer, Heidelberg (1995)

[6] Bellare, M., Rogaway, P.: Robust computational secrete sharing and a unified account of classical secret-sharing goals. In: 14th ACM Conference on Computer and Communications Security, Proceedings of CCS 2007, pp. 172–184. ACM Press, New York (2007)

[7] Bellare, M., Yilek, S.: Encryption schemes secure under selective opening attack. IACR ePrint Archive (2009)

[8] Ben-Or, M., Goldwasser, S., Wigderson, A.: Completeness theorems for non-cryptographic fault-tolerant distributed computation. In: 20th ACM Symposium on Theory of Computing, Proceedings of STOC 1988, pp. 1–10. ACM, New York (1988)

[9] Blum, M.: Coin flipping by telephone. In: Gersho, A. (ed.) Advances in Cryptology, A report on CRYPTO 1981, number 82-04 in ECE Report, pp. 11–15. University of California, Electrical and Computer Engineering (1982)

[10] Boldyreva, A., Fehr, S., O'Neill, A.: On notions of security for deterministic encryption, and efficient constructions without random oracles. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 335–359. Springer, Heidelberg (2008)

[11] Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. In: 42nd Annual Symposium on Foundations of Computer Science, Proceedings of FOCS 2001, pp. 136–145. IEEE Computer Society, Los Alamitos (2001)

[12] Canetti, R., Fischlin, M.: Universally composable commitments. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 19–40. Springer, Heidelberg (2001)

[13] Canetti, R., Feige, U., Goldreich, O., Naor, M.: Adaptively secure multi-party computation. In: Twenty-Eighth Annual ACM Symposium on Theory of Computing, Proceedings of STOC 1995, pp. 639–648. ACM Press, New York (1996)

[14] Canetti, R., Dwork, C., Naor, M., Ostrovsky, R.: Deniable encryption. In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 90–104. Springer, Heidelberg (1997)

[15] Canetti, R., Kilian, J., Petrank, E., Rosen, A.: Concurrent zero-knowledge requires $\tilde{\Omega}(\log n)$ rounds. In: 33rd Annual ACM Symposium on Theory of Computing, Proceedings of STOC 2001, pp. 570–579. ACM Press, New York (2001)
[16] Canetti, R., Halevi, S., Katz, J.: Adaptively-secure, non-interactive public-key encryption. In: Kilian, J. (ed.) TCC 2005. LNCS, vol. 3378, pp. 150–168. Springer, Heidelberg (2005)
[17] Chaum, D., Crépeau, C., Damgård, I.: Multiparty unconditionally secure protocols. In: 20th ACM Symposium on Theory of Computing, Proceedings of STOC 1988, pp. 11–19. ACM Press, New York (1988)
[18] Damgård, I., Nielsen, J.B.: Improved non-committing encryption schemes based on general complexity assumptions. In: Bellare, M. (ed.) CRYPTO 2000. LNCS, vol. 1880, pp. 432–450. Springer, Heidelberg (2000)
[19] Damgård, I.B., Pedersen, T.P., Pfitzmann, B.: On the existence of statistically hiding bit commitment schemes and fail-stop sigantures. In: Stinson, D.R. (ed.) CRYPTO 1993. LNCS, vol. 773, pp. 250–265. Springer, Heidelberg (1994)
[20] Dodis, Y., Oliveira, R., Pietrzak, K.: On the generic insecurity of the full domain hash. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 449–466. Springer, Heidelberg (2005)
[21] Dolev, D., Dwork, C., Naor, M.: Non-malleable cryptography. In: Twenty-Third Annual ACM Symposium on Theory of Computing, Proceedings of STOC 1991, pp. 542–552. ACM Press, New York (1991) (Extended abstract)
[22] Dwork, C., Naor, M., Reingold, O., Stockmeyer, L.: Magic functions. Journal of the ACM 50(6), 852–921 (2003)
[23] Dwork, C., Naor, M., Sahai, A.: Concurrent zero-knowledge. Journal of the ACM 51(6), 851–898 (2004)
[24] Gennaro, R., Micali, S.: Independent zero-knowledge sets. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4052, pp. 34–45. Springer, Heidelberg (2006)
[25] Goldreich, O.: Foundations of Cryptography (Basic Tools), vol. 1. Cambridge University Press, Cambridge (2001)
[26] Goldreich, O., Krawczyk, H.: On the composition of zero-knowledge proof systems. SIAM Journal on Computing 25(1), 169–192 (1996)
[27] Goldwasser, S., Micali, S.: Probabilistic encryption. Journal of Computer and System Sciences 28(2) (1984)
[28] Haitner, I., Holenstein, T.: On the (im)possibility of key dependent encryption. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 202–219. Springer, Heidelberg (2009)
[29] Haitner, I., Reingold, O.: Statistically-hiding commitment from any one-way function. In: 39th Annual ACM Symposium on Theory of Computing, Proceedings of STOC 2007, pp. 1–10. ACM Press, New York (2007)
[30] Haitner, I., Hoch, J.J., Reingold, O., Segev, G.: Finding collisions in interactive protocols – a tight lower bound on the round complexity of statistically-hiding commitments. In: 48th Annual Symposium on Foundations of Computer Science, Proceedings of FOCS 2007, pp. 669–679. IEEE Computer Society, Los Alamitos (2007)
[31] Hofheinz, D.: Possibility and impossibility results for selective decommitments. IACR ePrint Archive (April 2008)
[32] Impagliazzo, R., Rudich, S.: Limits on the provable consequences of one-way permutations. In: Twenty-First Annual ACM Symposium on Theory of Computing, Proceedings of STOC 1989, pp. 44–61. ACM Press, New York (1989) (Extended abstract)

[33] Kilian, J., Petrank, E.: Concurrent and resettable zero-knowledge in poly-logarithmic rounds. In: 33rd Annual ACM Symposium on Theory of Computing, Proceedings of STOC 2001, pp. 560–569. ACM Press, New York (2001)
[34] Kol, G., Naor, M.: Cryptography and game theory: Designing protocols for exchanging information. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 320–339. Springer, Heidelberg (2008)
[35] Naor, M.: Bit commitment using pseudo-randomness. Journal of Cryptology 4(2), 151–158 (1991)
[36] Naor, M., Pinkas, B.: Efficient oblivious transfer protocols. In: Twelfth Annual Symposium on Discrete Algorithms, Proceedings of SODA 2001, pp. 448–457. ACM/SIAM (2001)
[37] Naor, M., Yung, M.: Universal one-way hash functions and their cryptographic applications. In: Twenty-First Annual ACM Symposium on Theory of Computing, Proceedings of STOC 1989, pp. 33–43. ACM Press, New York (1989)
[38] Nielsen, J.B.: Separating random oracle proofs from complexity theoretic proofs: The non-committing encryption case. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 111–126. Springer, Heidelberg (2002)
[39] Panjwani, S.: Tackling adaptive corruptions in multicast encryption protocols. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 21–40. Springer, Heidelberg (2007)
[40] Peikert, C., Waters, B.: Lossy trapdoor functions and their applications. In: Fotieth Annual ACM Symposium on Theory of Computing, Proceedings of STOC 2008, pp. 187–196. ACM Press, New York (2008)
[41] Peikert, C., Vaikuntanathan, V., Waters, B.: A framework for efficient and composable oblivious transfer. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 554–571. Springer, Heidelberg (2008)
[42] Prabhakaran, M., Rosen, A., Sahai, A.: Concurrent zero knowledge with logarithmic round complexity. In: 43rd Annual Symposium on Foundations of Computer Science, Proceedings of FOCS 2002, pp. 366–375. IEEE Computer Society Press, Los Alamitos (2002)
[43] Reingold, O., Trevisan, L., Vadhan, S.P.: Notions of reducibility between cryptographic primitives. In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 1–20. Springer, Heidelberg (2004)
[44] Richardson, R., Kilian, J.: On the concurrent composition of zero-knowledge proofs. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 415–431. Springer, Heidelberg (1999)
[45] Rosen, A., Segev, G.: Efficient lossy trapdoor functions based on the composite residuosity assumption. IACR ePrint Archive (March 2008)
[46] Simon, D.R.: Findings collisions on a one-way street: Can secure hash functions be based on general assumptions? In: Nyberg, K. (ed.) EUROCRYPT 1998. LNCS, vol. 1403, pp. 334–345. Springer, Heidelberg (1998)
[47] Wee, H.M.: One-way permutations, interactive hashing and statistically hiding commitments. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 419–433. Springer, Heidelberg (2007)