A Unified Methodology For Constructing Public-Key Encryption Schemes Secure Against Adaptive Chosen-Ciphertext Attack

EDITH ELKIND AMIT SAHAI*

Abstract

We introduce a new methodology for achieving security against adaptive chosen-ciphertext attack (CCA) for public-key encryption schemes, which we call the *oblivious decryptors model*. The oblivious decryptors model generalizes both the two-key model of Naor and Yung, as well the Cramer–Shoup encryption schemes. The key ingredient in our new paradigm is Sahai's notion of Simulation-Sound NIZK proofs.

Our methodology is easy to use: First, construct an encryption scheme which satisfies the "bare" oblivious-decryptors model: This can be done quite easily, with simple proofs of security. Then, by adding a Simulation-Sound NIZK proof, the scheme becomes provably CCA-secure. Note that this paradigm allows for the use of *efficient* special-purpose Simulation-Sound NIZK proofs, such as those recently put forward by Cramer and Shoup. We also show how to present all known efficient (provably secure) CCA-secure public-key encryption schemes as special cases of our model.

Keywords: Public-Key Encryption, Chosen-Ciphertext Security, Zero Knowledge, Non-Interactive Zero Knowledge, Non-Malleability

* Dept of Computer Science, Princeton University. Princeton, NJ 08544. E-Mail: `elkind, sahai@cs.princeton.edu`.

1 Introduction

Achieving provable chosen-ciphertext security [12, 13] for public-key encryption has been one of the main challenges for cryptographic research of the past several years. The first public-key encryption scheme provably secure against adaptive chosen-ciphertext attack was given in the pioneering work of Dolev, Dwork, and Naor [6]. In the last few years, two additional results have significantly informed our understanding of provable chosen-ciphertext security. The first was the Cramer–Shoup cryptosystem [3], which yielded the first practical adaptive CCA-secure scheme based on a specific algebraic assumption. The second was the introduction by Sahai [14] of the notion of *Simulation-Sound Non-Interactive Zero Knowledge (NIZK)*, and the proof that the simple two-key paradigm of Naor and Yung [12] can be adapted to yield adaptive CCA-security by adding a simulation-sound NIZK proof of consistency of two ciphertexts. Such a two-key scheme could be based on general assumptions, and it was hoped that because of its simplicity, this approach would yield practical schemes, as well.

These two works, however, used seemingly very different techniques. Although a number of researchers have observed that a portion of the Cramer–Shoup cryptosystem can be seen as a Simulation-Sound NIZK proof system, the rest of the cryptosystem, which has been generalized in [5], does not fit into the two-key paradigm. The two-key paradigm and the proof of security given in [14] make critical use of the existence of two decryption keys, which allow for a reduction of the security of the overall scheme to the passive security of a component scheme. The proof of [14] furthermore relies on standard computational indistinguishability arguments. Unfortunately, no practical instantiation of the two-key scheme has ever been given, because of the difficulty of designing NIZK proofs of consistency of encryptions for two different public keys.

The Cramer–Shoup cryptosystem [3] and its generalizations [5], on the other hand, make use of only a single decryption key. Furthermore, their proof of security does not reduce attacks on the scheme to an attack on a simpler encryption scheme, but rather argues directly that a successful attack would contradict some underlying assumption, such as the Decisional Diffie-Hellman assumption in [3]. Also, the proof of security uses statistical arguments in a crucial way[1].

Our Contribution. In this work, we propose a new model for generating public-key encryption schemes secure against adaptive chosen-ciphertext attack, which we call the *oblivious decryptors* model. This model is easy to apply, and can yield efficient public-key encryption schemes. Indeed, our model unifies the work of Sahai [14] and Cramer and Shoup [3, 5], as it generalizes both the two-key model, as well as the Cramer–Shoup cryptosystem and its generalizations. This model makes use of Simulation-Sound NIZK proofs to yield adaptive CCA-secure public-key encryption schemes, which in many cases can be quite practical.

Our model gives rise to a new design methodology for constructing provably-secure public-key encryption schemes secure against adaptive chosen-ciphertext attack: First, construct an encryption scheme which satisfies the "bare" oblivious-decryptors model. It turns out that this can be done quite easily, and both two-key encryptions *without the NIZK proof* as well as "Cramer–Shoup Lite" can be shown to satisfy this definition with simple, standard arguments. Note that this first part does not involve dealing with any kind of chosen-ciphertext attacks. Then, if one adds a Simulation-Sound NIZK proof of "well-formedness" of the encryption, our proof immediately implies that the resulting scheme is adaptive CCA-secure.

We stress that our model is able to incorporate both computational and statistical indistinguishability arguments. Somewhat surprisingly, this robustness of our model sheds light on the connection between the two-key paradigm and the Cramer–Shoup cryptosystems. The Cramer–Shoup cryptosystems somehow are able to make do with only one decryption key, whereas the two-key paradigm, as the name implies, needs two. What our model shows is that the Cramer–Shoup cryptosystems essentially *replace* the second key from the two-key paradigm with a statistical indistinguishability argument. The oblivious decryptors model

[1]This is, of course, quite commonplace in cryptographic constructions based on algebraic assumptions as in [3, 5]. But as we shall see, this use of statistical arguments turns out to be a critical point that we address.

formalizes this in a precise way.

The primacy of Simulation-Sound NIZK. Our work shows that the notion of Simulation-Sound NIZK can be used to unify recent work on adaptive CCA-secure public-key encryption, and points out the importance of Simulation-Soundness in all known practical provably-secure (without random oracles) such schemes.

We stress that because, in the oblivious-decryptors model, the Simulation-Sound NIZK proof is only of a "well-formedness" condition, and not (necessarily) of multiple-encryption consistency as in [14], it is possible that efficient special-purpose Simulation-Sound NIZK proofs, such as those based on Universal Projective Hashing [5], can be used. Indeed, we show that it is sufficient, for example, to have Simulation-Sound NIZK proofs for essentially any hard-on-average problem which admits a hash proof system of [5] to be based on it. Thus, the efficient Simulation-Sound NIZK proofs given in [4, 5] for such problems as Discrete-Log Equality, Quadratic Residuosity, and a problem related to Pallier's scheme, immediately give rise to efficient CCA-secure public-key encryption schemes through the oblivious-decryptors model, which coincide with some of the schemes presented in [5], and are quite practical.

Since Simulation-Sound NIZK proves to be so important in this context, one may hope that perhaps the techniques put forward by Cramer and Shoup [4, 5] based on Universal Projective Hashing could be used to construct efficient Simulation-Sound NIZK Proofs for all NP languages. Unfortunately, we show that any proof system, whether non-interactive or not, which is based on Universal Projective Hashing must necessarily be limited to problems which admit Statistical Zero-Knowledge Proofs. This implies that if any NP-complete language admits a proof based on Universal Projective Hashing, then the polynomial-time hierarchy must collapse.

2 Preliminaries and Definitions

We use standard notations and conventions for writing probabilistic algorithms and experiments. If A is a probabilistic algorithm, then $A(x_1, x_2, \ldots; r)$ is the result of running A on inputs $x_1, x_2, \ldots$ and coins r. We let $y \leftarrow A(x_1, x_2, \ldots)$ denote the experiment of picking r at random and letting y be $A(x_1, x_2, \ldots; r)$. If S is a finite set then $x \leftarrow S$ is the operation of picking an element uniformly from S. $x := \alpha$ is a simple assignment statement. By a "non-uniform probabilistic polynomial-time adversary," we always mean a circuit whose size is polynomial in the security parameter.

The specific hard languages we consider are often parametrized by information known to all parties. To formalize this idea, we consider a language L that consists of pairs of the form (σ, x) and set $L_\sigma = \{x \mid (\sigma, x) \in L\}$; if Rel is the witness relation that corresponds to L, we set $Rel_\sigma = \{(x, w) \mid ((\sigma, x), w) \in Rel\}$.

We make use of the notion of designated-verifier NIZK, which captures the following intuition: in certain circumstances, it is safe to allow the verifier to choose the public reference string used for NIZK; in this case, the verifier can also generate some private information that is necessary to check the correctness of the proof. Consequently, the proofs cannot be verified by the general public, but by the designated verifier only. Designated-verifier NIZK may be easier to implement than general NIZK and is sufficient for our purposes.

Definition 2.1 [designated-verifier NIZK] $\Pi = (G^{NIZK}, \mathrm{P}, \mathcal{V}, \mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2))$ is a *single-theorem simulation-sound designated-verifier NIZK proof system* for the language $L \in \mathrm{NP}$ with witness relation Rel if G^{NIZK}, P,$\mathcal{V}$,$\mathcal{S}_1$,$\mathcal{S}_2$ are all probabilistic polynomial-time machines and there exists a negligible function $\alpha(k)$ such that for all k:

(Completeness): For any σ, all $x \in L_\sigma$ and all w such that $Rel_\sigma(x, w) = \texttt{true}$, for all pairs (R, ρ) produced by $G^{NIZK}(\sigma)$, we have that $\mathcal{V}(x, \mathrm{P}(x, w, R), \rho) = \texttt{true}$.

(Soundness): For all probabilistic polynomial-time adversaries A, if (R, ρ) is produced by $G^{NIZK}(\sigma)$, then the probability that $A(R)$ will output (x, π) such that $x \notin L_\sigma$ but $\mathcal{V}(x, \pi, \rho) = \texttt{true}$ is less than $\alpha(k)$.

(Single-Theorem Adaptive Zero-Knowledge): For all non-uniform probabilistic polynomial-time adversaries $A = (A_1, A_2)$ and for all σ we have that $|\Pr[\mathsf{Expt}_A(\sigma) = 1] - \Pr[\mathsf{Expt}_A^S(\sigma) = 1]| \leq \alpha(k)$, where the experiments $\mathsf{Expt}_A(\sigma)$ and $\mathsf{Expt}_A^S(\sigma)$ are defined as follows:

$\mathsf{Expt}_A(\sigma)$:	$\mathsf{Expt}_A^S(\sigma)$:
$(R, \rho) \leftarrow G^{NIZK}(\sigma)$	$(R, \rho, \mu) \leftarrow \mathcal{S}_1(\sigma)$
$(x, w, \xi) \leftarrow A_1(\rho)$	$(x, w, \xi) \leftarrow A_1(\rho)$
$\pi \leftarrow P(x, w, R)$	$\pi \leftarrow \mathcal{S}_2(x, \mu)$
$g \leftarrow A_2(\pi, \xi)$	$g \leftarrow A_2(\pi, \xi)$

(One-time Simulation Soundness): For all non-uniform probabilistic polynomial-time adversaries $A = (A_1, A_2)$, we have that $\Pr[\mathsf{Expt}_{A,\Pi}(\sigma) = 1] \leq \alpha(k)$, where the $\mathsf{Expt}_{A,\Pi}(\sigma)$ is the following experiment:

$\mathsf{Expt}_{A,\Pi}(\sigma)$:
  $(R, \rho, \mu) \leftarrow \mathcal{S}_1(\sigma)$
  $(x', \xi) \leftarrow A_1^{\mathcal{V}(\cdot,\cdot,\rho)}(\sigma, R)$
  $\pi' \leftarrow \mathcal{S}_2(x', \mu)$
  $(x, \pi) \leftarrow A_2^{\mathcal{V}(\cdot,\cdot,\rho)}(\pi', \xi)$
  `return true` iff
    $(x, \pi) \neq (x', \pi')$ `and`
    $x \notin L_\sigma$ `and`
    $\mathcal{V}(x, \pi, \rho) = \texttt{true}$

Remark 2.2 Note that in the simulation-soundness experiment, the adversary is given access to the verification oracle; this reflects the fact that in the designated verifier model, the adversary cannot check the validity of the proofs on its own, so we must provide him with a way of doing that.

Remark 2.3 This definition is stronger than the one given in [S], although it is achieved by their construction. Namely, in addition to the simulation-soundness in the sense of [S], it also guarantees us computational unique applicability: it is computationally difficult for the adversary to find a new theorem for which the proof given by the simulator is valid. Also, it must be hard for the adversary to find a new proof for the theorem proved by the simulator (note that the latter requirement is not implied by soundness, since the common reference string is produced by S_1 instead of G^{NIZK}).

Remark 2.4 We can also define NIZK proof systems with full simulation soundness and many-theorem zero-knowledge for the designated-verifier model; the changes to the original defintions are straightforward.

We also use the standard definition of encryption scheme secure against adaptive chosen-ciphertext attack.

Definition 2.5 We say that an encryption scheme (G, E, D) is secure against adaptive chosen-ciphertext attack if for any nonuniform probabilistic polynomial-time adversary $A = (A_1, A_2)$ there is a negligible function $\alpha(k)$ such that for all k $\left|\Pr[\mathsf{Expt}_A^{CCA}(k) = 1] - 1/2\right| \leq \alpha(k)$, where $\mathsf{Expt}_A^{CCA}(k)$ is the following experiment:

$$
\begin{array}{l}
\mathsf{Expt}_A^{CCA}(k): \\
\quad (pk, sk) \leftarrow G(1^k) \\
\quad (m_0, m_1, \tau) \leftarrow A_1^{D(sk,\cdot)}(pk) \\
\quad b \leftarrow \{0,1\} \\
\quad c \leftarrow E(pk, m_b) \\
\quad g \leftarrow A_2^{D'(sk,\cdot)}(c, \tau) \\
\quad \texttt{return } 1 \text{ iff } g = b
\end{array}
$$

Above, D' stands for the oracle that decrypts any ciphertext except c; also, we require that A_1 outputs m_0, m_1 with $|m_0| = |m_1|$.

3 Oblivious Decryptors Model

We define the notion of *oblivious decryptors model*, which generalizes some of the previously known schemes secure against adaptive CCA attacks, and prove that the encryption scheme that can be constructed based on it is itself secure against this kind of attack.

3.1 ODM - Intuition

Essentially, an oblivious decryptors encryption scheme is an ordinary encryption scheme augmented with a pair of "alternative" decryption oracles, which always produce the correct result on well-formed ciphertexts – but whose behavior on invalid ciphertexts is unconstrained. The security guarantees of this scheme come in the form of indistinguishability conditions related to these oracles. Namely, we require that:

- An efficient adversary that only has access to the first oracle cannot distinguish a well-formed ciphertext from an invalid ciphertext, which is produced by an invalid ciphertext sampling algorithm (which is a part of the scheme).
- An efficient adversary that only has access to the second oracle has no significant advantage in distinguishing an *invalid* "encryption" of one message from an invalid "encryption" of another message.

Note that, generally speaking, these oracles are not able to verify if the ciphertext they received is well-formed; indeed, nothing can be said about their behavior if this is not the case, and this is why we call them *oblivious* decryptors.

These two decryption oracles do not have to be efficiently implementable, as they are not used for actual encryption/decryption; rather, they are used in a hybrid argument that demonstrates the adaptive security of the encryption scheme when combined with simulation-sound NIZK proofs. In this scheme, the sender is required to provide a simulation-sound NIZK proof that the ciphertext it outputs is well-formed. The reason for this requirement is as follows: the validity of the adaptive CCA adversary's queries to its decryption oracle can be checked by simply checking the proof; once the validity of the query ciphertext has been confirmed, it does not matter whether the adversary has access to an actual decryption oracle or to one of the alternative ones – therefore, we can use the indistinguishablility conditions above to complete our proof of security for adaptive CCA.

At first glance, it may seem that the indistinguishability conditions above are as problematic to establish as proving chosen-ciphertext security. In fact, these conditions are quite easy to satisfy. To illustrate this point, we give two examples in the next section. Hence, finding an encryption scheme with two "oblivious" oracles that fulfill these conditions, and a suitable simulation-sound NIZK proof system, can be a much easier task than constructing an adaptively CCA-secure encryption scheme from scratch.

3.2 ODM - Definition

In the following definition, the generation algorithm is broken down into two phases. During the first stage, the public parameter generation algorithm G^σ takes 1^k as an input and produces σ; during the second stage, G takes σ as an input and produces a public key, a secret key, and other data. The reason for this separation is outlined in the previous section.

Definition 3.1 [Oblivious Decryptors Encryption Scheme] An *oblivious decryptors encryption scheme* is an encryption scheme (G, E, D) together with algorithms $G^\sigma, \overline{E}, \mathcal{D}_1, \mathcal{D}_2$ and a language L that satisfies the following conditions:

- G^σ, G, E, D, $\overline{E}$ are all probabilistic polynomial-time machines.
- On input 1^k, algorithm G^σ outputs σ.
- On input σ, algorithm G outputs bit strings (pk, sk, h, d_1, d_2).
- L is a language [2] in NP with witness relation $Rel = \{(pk, c), (m, r) \mid c = E(pk, m; r)\}$.
- For any σ and any tuple (pk, sk, h, d_1, d_2) in the range of $G(\sigma)$ and for any message m, $|m| = k$, it holds that $D(sk, E(pk, m)) = m$.
- For any σ and any tuple (pk, sk, h, d_1, d_2) in the range of $G(\sigma)$, it holds that $(pk, c) \in L$ implies $D(sk, c) = \mathcal{D}_1(d_1, c) = \mathcal{D}_2(d_2, c)$.
- For all non-uniform probabilistic polynomial-time adversaries $A = (A_1, A_2)$ there exists a negligible function $\alpha(k)$ such that $|p_0(b) - p_1(b)| \leq \alpha(k)$ for $b = 0, 1$, where $p_0(b) = \Pr\left[\mathsf{ODMExpt}^0(b) = 1\right]$, $p_1(b) = \Pr\left[\mathsf{ODMExpt}^1(b) = 1\right]$, and the experiments $\mathsf{ODMExpt}^0(b)$ and $\mathsf{ODMExpt}^1(b)$ are defined as follows:

$\mathsf{ODMExpt}^0(b)$: $\sigma \leftarrow G^\sigma(1^k)$ $(pk, sk, h, d_1, d_2) \leftarrow G(\sigma)$ $(m_0, m_1, \alpha) \leftarrow A_1^{\mathcal{D}_1(d_1,\cdot)}(pk)$ $c \leftarrow E(pk, m_b)$ $\texttt{return}\ A_2^{\mathcal{D}_1(d_1,\cdot)}(c, \alpha)$
$\mathsf{ODMExpt}^1(b)$: $\sigma \leftarrow G^\sigma(1^k)$ $(pk, sk, h, d_1, d_2) \leftarrow G(\sigma)$ $(m_0, m_1, \alpha) \leftarrow A_1^{\mathcal{D}_1(d_1,\cdot)}(pk)$ $c \leftarrow \overline{E}(h, m_b)$ $\texttt{return}\ A_2^{\mathcal{D}_1(d_1,\cdot)}(c, \alpha)$

- For all non-uniform probabilistic polynomial-time adversaries $A = (A_1, A_2)$, there exists a negligible function $\alpha(k)$ such that $p_2(0) - p_2(1) \leq \alpha(k)$, where $p_2(b) = \Pr\left[\mathsf{ODMExpt}^2(b) = 1\right]$, and the experiment $\mathsf{ODMExpt}^2(b)$ is defined as follows:

[2] For the purposes of our construction, we can replace L with any L', $L \subseteq L'$ provided that the other conditions in this definition are satisfied with respect to L'.

$\mathsf{ODMExpt}^2(b)$:
$\sigma \leftarrow G^\sigma(1^k)$
$(pk, sk, h, d_1, d_2) \leftarrow G(\sigma)$
$(m_0, m_1, \xi) \leftarrow A_1^{\mathcal{D}_2(d_2,\cdot)}(pk)$
$c \leftarrow \overline{E}(h, m_b)$
$\texttt{return}\ A_2^{\mathcal{D}_2(d_2,\cdot)}(c, \xi)$

Remark 3.2 We do not require $\mathcal{D}_1$, $\mathcal{D}_2$ to be efficient; for example, it will be the case in one of our constructions that $\mathcal{D}_2$ is an all-powerful machine; for such $\mathcal{D}_2$, d_2 may consist of public data.

Remark 3.3 It may look like these requirements by themselves are just as strong as adaptive CCA security; this impression is not correct because there are no guarantees on the behavior of $\mathcal{D}_1, \mathcal{D}_2$ when they are asked to decrypt something that is not a valid encryption, as may be the case in $\mathsf{ODMExpt}^2$. These issues have to be addresed separately, and it is exactly here that simulation-sound NIZK comes to rescue. Indeed, the way that these requirements can be met is by designing decryptors $\mathcal{D}_1$ and $\mathcal{D}_2$ which do not in any way interfere with the indistinguishability argument. See the example of the two-key paradigm that follows next to see how this works.

4 Examples of Oblivious Decryptors Schemes

Before we delve into adaptive CCA security, let us see two examlpes of schemes which fit into the Oblivious Decryptors Model, based on previous work:

Example 4.1 Here, we show how two-key encryption [12, 14] fits into this model. An instance of an oblivious decryptors model based on a semantically secure encryption scheme (G_0, E_0, D_0) can be constructed as follows (note that when we apply to this scheme the adaptive CCA compiler described in the next section, we obtain the adaptive CCA-secure encryption scheme of [14]).

$G^\sigma(1^k)$(Public Parameter Generation) :
$\texttt{return}\ 1^k$

$G(\sigma)$(Key Generation) :
$(e_1, d_1) \leftarrow G_0(1^k)$
$(e_2, d_2) \leftarrow G_0(1^k)$
$pk = (e_1, e_2), sk = (d_1, d_2)$
$h = pk$
$\texttt{return}\ (pk, sk, h, d_1, d_2)$

$E(pk, m)$(Encryption) :
$c_1 = E_0(e_1, m), c_2 = E_0(e_2, m)$
$\texttt{return}\ (c_1, c_2)$

$D(sk, c_1, c_2)$(Decryption) :
$\texttt{return}\ D_0(d_1, c_1)$ or $D_0(d_2, c_2)$, chosen arbitrarily.

$\mathcal{D}_i(d_i, c_1, c_2)$(Decryption Oracles) :
$\texttt{return}\ D_0(d_i, c_i)$

$\overline{E}(h, m_b)$(Invalid Ciphertext Sampling Algorithm) :
$c_1 = E_0(e_1, m_b), c_2 = E_0(e_2, 0^k)$
$\texttt{return}\ (c_1, c_2)$

We set L to be the language of all proper encryptions, that is, $L = \{(e_1, e_2), c_1, c_2) \mid \exists m, r_1, r_2 : c_1 = E(m, e_1; r_1), c_2 = E(m, e_2; r_2)\}$.

There are two indistinguishability conditions to be checked.

The difference between $\mathsf{ODMExpt}^0(b)$ and $\mathsf{ODMExpt}^1(b)$ in this case is that in one of them the ciphertext is $(E(e_1, m_b), E(e_2, m_b))$ whereas in the other one the ciphertext is $(E(e_1, m_b), E(e_2, 0^k))$. However, in both cases the decryption oracle only uses d_1 to decrypt the queries, so the indistinguishability of these experiments reduces to semantic security of (the second instance of) the original encryption scheme.

In the second pair of experiments, the ciphertext is $(E(e_1, m_0), E(e_2, 0^k))$ and $(E(e_1, m_1), E(e_2, 0^k))$, respectively. Again, the decryption oracle only uses d_2 to decrypt the queries, so the indistinguishability of these experiments reduces to semantic security of (the first instance of) the original encryption scheme.

Example 4.2 A highly efficient example of oblivious decryptors model can be based on Decisional Diffie-Hellman Assumption. When combined with simulation-sound NIZK proofs, this yields the encryption scheme [3] of [3].

$G^\sigma(1^k)$(Public Parameter Generation) :
  Choose a group G, $|G| = p$, $|p| = k$, p is a prime
  so that DDH holds for G
  Choose two generators $g_1, g_2 \in G$
  Set $\sigma = (p, g_1, g_2)$
  Output σ

$G(\sigma)$(Key Generation) :
  Choose $z_1, z_2 \in \mathbb{Z}_p$
  Set $f = g_1^{z_1} g_2^{z_2}$
  Set $pk = (f, \sigma)$, $sk = (z_1, z_2, \sigma)$
  Set $h = sk$, $d_1 = sk$, $d_2 = pk$
  `return` (pk, sk, h, d_1, d_2)

$E(pk, m)$(Encryption) :
  Choose $w \in \mathbb{Z}_p$
  Set $x_1 = g_1^w$, $x_2 = g_2^w$, $y = f^w \cdot m$
  `return` (x_1, x_2, y)

$D(sk, (x_1, x_2, y))$(Decryption) :
  `return` $x_1^{-z_1} x_2^{-z_2} y$

$\mathcal{D}_1(d_1, (x_1, x_2, y))$(First Decryption Oracle) :
  `return` $x_1^{-z_1} x_2^{-z_2} y$

$\mathcal{D}_2(d_2, (x_1, x_2, y))$(Second Decryption Oracle) :
  Set $w' = \log_{g_1} x_1$ (Note this need *not* be efficient!)
  `return` $f^{-w'} y$

$\overline{E}(h, m_b)$(Invalid Ciphertext Sampling Algorithm) :
  Choose $w, w' \in \mathbb{Z}_p$
  Set $x_1 = g_1^w$, $x_2 = g_2^{w'}$, $y = x_1^{z_1} x_2^{z_2} m_b$
  `return` (x_1, x_2, y)

Also, we set $L = \{((p, g_1, g_2), (x, y)) \mid \langle g_1 \rangle = \langle g_2 \rangle = G, |G| = p, x, y \in G, \exists w : x = g_1^w, y = g_2^w\}$.

It is easy to see that the correctness conditions ($D(sk, E(pk, m)) = m$, $D(sk, c) = \mathcal{D}_1(d_1, c) = \mathcal{D}_2(d_2, c)$ for suitable values of the parameters involved) are satisfied.

[3] The proof of adaptive CCA security in [3] uses a modified version of the protocol, which is proved to be equivalent to the original one, but slightly less efficient; the output of our compiler is this modified protocol.

The first of the indistinguishablity conditions follows immediately from DDH. The second indistinguishability condition is implied by the fact that the pairs $(f = g_1^{z_1} g_2^{z_2}, \overline{E}(h, m_0))$ and $(f = g_1^{z_1} g_2^{z_2}, \overline{E}(h, m_1))$ are *identically* distributed. Therefore, regardless of the computational power of the adversary together with $\mathcal{D}_2$, it cannot distinguish these statistically indistinguishable distributions.

5 Adaptive CCA Security

For adaptive CCA security, we will extend the encryption scheme that can be extracted from an oblivious decryptors scheme by requiring the sender to provide simulation-sound designated-verifier NIZK proofs of $(pk, c) \in L$ using the message m (and the random coins used for encryption) as a witness.

$\hat{G}(1^k)$(Key Generation) :
    $\sigma \leftarrow G^{\sigma}(1^k)$
    $(pk, sk, h, d_1, d_2) \leftarrow G(\sigma)$
    $(R, \rho) \leftarrow G^{NIZK}(\sigma)$
    $PK = (pk, R), SK = (sk, \rho)$

$\hat{E}(PK, m)$(Encryption) :
    $c = E(pk, m; r)$
    $\pi = \mathrm{P}(c, (m, r), R)$
    `return` (c, π)

$\hat{D}(SK, c, \pi)$(Decryption) :
    `if` $\mathcal{V}(c, \pi, \rho)$ = `reject then abort`
    `else return` $D(sk, c)$.

Here G^{NIZK}, P, $\mathcal{V}$ are as in the definition of designated-verifier NIZK.

5.1 Adaptive CCA Security - Proof

Suppose that there is an adaptive CCA adversary that can break this scheme. We will use a hybrid argument to show that this contradicts one of the properties of the oblivious decryptors model. Namely, we will construct distinguishers that have nonnegligible success probability in the experiments described in the definition of the oblivious decryptors scheme.

Assume that there exists a nonuniform probabilistic polynomial-time adversary $A = (A_1, A_2)$ that can achieve a non-negligible advantage $\epsilon(k)$ in an adaptive CCA attack on this scheme, that is, $|p_0 - p_9| > \epsilon(k)$, where $p_i = \Pr[\mathsf{Expt}_i = 1]$, $i = 0, \ldots, 9$, and $\mathsf{Expt}_0, \ldots, \mathsf{Expt}_9$ are defined below. We will show that $|p_{i+1} - p_i|$, $i = 0, \ldots, 8$ are all negligible and thus arrive to a contradiction.

We say that a ciphertext (c, π) is *proper* with respect to a public key $PK = (pk, R)$ if $(pk, c) \in L$. Furthermore, we say that a ciphertext (c, π) is *valid* with respect to a private key $SK = (sk, \rho)$ if $\mathcal{V}(c, \pi, \rho) =$ `true` . The values of PK, SK will usually be clear from the context. For any pair (PK, SK), it holds that if (c, π) is proper with respect to PK, then (c, π) is valid with respect to SK, but not vice versa.

Expt_0 :
  Set up PK, SK:
    $\sigma \leftarrow G^{\sigma}(1^k)$
    $(pk, sk, h, d_1, d_2) \leftarrow G(\sigma)$
    $(R, \rho) \leftarrow G^{NIZK}(\sigma)$
    $PK = (pk, R), SK = (sk, \rho)$
  $(m_0, m_1, \alpha) \leftarrow A_1^{D(SK,\cdot)}(PK)$
  Set up challenge:
    $c := E(pk, m_0; r)$
    $\pi := \mathrm{P}(c, (m_0, r), R)$
  $\texttt{return}\ A_2^{D'(SK,\cdot)}(c, \pi, \alpha)$

Note that the decryption oracle D' rejects if given (c, π).

In what follows, we mark with an (*) the lines that were changed compared to the previous experiment.

Expt_1 :
  Set up PK, SK:
    $\sigma \leftarrow G^{\sigma}(1^k)$
    $(pk, sk, h, d_1, d_2) \leftarrow G(\sigma)$
    $(R, \rho, \gamma) \leftarrow S_1(\sigma)$ (*)
    $PK = (pk, R), SK = (sk, \rho)$
  $(m_0, m_1, \alpha) \leftarrow A_1^{D(SK,\cdot)}(PK)$
  Set up challenge:
    $c := E(pk, m_0)$
    $\pi := \mathcal{S}_2(c, R, \gamma)$ (*)
  $\texttt{return}\ A_2^{D'(SK,\cdot)}(c, \pi, \alpha)$

The only difference between Expt_0 and Expt_1 is in using simulated NIZK proofs instead of actual proofs; no efficient adversary can notice this with non-negligible probability, so $|p_0 - p_1|$ is negligible.

Expt_2 :
  Set up PK, SK:
    $\sigma \leftarrow G^{\sigma}(1^k)$
    $(pk, sk, h, d_1, d_2) \leftarrow G(\sigma)$
    $(R, \rho, \gamma) \leftarrow S_1(\sigma)$
    $PK = (pk, R), SK = (sk, \rho)$
  $(m_0, m_1, \alpha) \leftarrow A_1^{\tilde{\mathcal{D}}_1(d_1,\rho,\cdot)}(PK)$ (*)
  Set up challenge:
    $c := E(pk, m_0)$
    $\pi := \mathcal{S}_2(c, R, \gamma)$
  $\texttt{return}\ A_2^{\tilde{\mathcal{D}}'_1(d_1,\rho,\cdot)}(c, \pi, \alpha)$ (*)

At this stage, we replace the standard decryption oracle with $\tilde{\mathcal{D}}_1$, which first runs $\mathcal{V}$ to check if it is given a valid ciphertext, rejects if this is not the case and calls $\mathcal{D}_1$ on the ciphertext otherwise. Note that if

A never asks a valid improper query or only asks to decrypt (c, π) during the second stage of the attack (in which case it is rejected by both oracles), $\tilde{\mathcal{D}}_1$ behaves identically to the ordinary decryption oracle. That is, there are only two situations in which replies of these oracles can differ: either A makes a valid but improper query different from (c, π), or A asks for a decryption of (c, π) during the first stage of the attack. We observe that the probability of the latter event is clearly negligible, as c is output by a semantically secure encryption algorithm. For this reason, in what follows we assume that it does not happen, as with overwhelming probability this is indeed the case. Hence, to show that $|p_2 - p_1|$ is negligible, it suffices to prove the following lemma.

Lemma 5.1 *In the setting of* $\mathsf{Expt}_1, \mathsf{Expt}_2$, *for any probabilistic polynomial-time adversary* $A = (A_1, A_2)$, *the probability that* A *will make a valid but improper query different from the challenge ciphertext* (c, π) *to the decryption oracle is negligible in* k.

Proof: We will use an adversary $A = (A_1, A_2)$ that can produce a valid improper ciphertext to construct an adversary $\hat{A} = (\hat{A}_1, \hat{A}_2)$ that has a nonnegligible advantage in the simulation-soundness experiment.

$\hat{A}_1(\sigma, R)$:
    Initialize $C := \emptyset$
    Set up PK, SK:
        $(pk, sk, h, d_1, d_2) \leftarrow G(\sigma)$
        $PK = (pk, R), SK = (sk, \rho)$
    Simulate first stage of attack:
        $(m_0, m_1, \alpha) \leftarrow A_1^{\hat{D}(sk,\cdot)}(PK)$
        store all queries of A_1 in C
    Set up challenge:
        $c := E(pk, m_0)$
        `return` $(c, (c, \alpha, \sigma, PK, SK))$

Here $\hat{A}_1$ implements the decryption oracle for A_1 by verifying the validity of the proof in the query (which $\hat{A}_1$ can do because he has access to the verification oracle) and decrypting the message using sk, which he generated by himself.

After $\hat{A}_1$ outputs $(x', \xi) = (c, (c, \alpha, \sigma, PK, SK))$, the simulator constructs a proof π' of $(pk, c) \in L_\sigma$ and passes it to $\hat{A}_2$.

$\hat{A}_2(\pi', (c, \alpha, \sigma, PK, SK))$:
    Simulate second stage of attack:
        $g \leftarrow A_2^{\hat{D}'(sk,\cdot)}(PK, \alpha)$
        add all queries of A_2 to C
    $c' \leftarrow C$
    `return` c'

Note that the size of C is at most polynomial in k, so if C contains a valid improper ciphertext, $\hat{A}$ will output it with nonnegligible probability. Hence, if the probability that at any point in the experiment, A produces a valid improper query $(c, \pi) \neq (c', \pi')$ is nonnegligible, the probability that $\hat{A}$ succeeds is nonnegligible as well, and we get a contradiction.

To complete the proof of the lemma, we have to show that if A makes a valid improper query in Expt_1 or Expt_2, then with overwhelming probability A will make such a query in our experiment as well. But this follows directly from the observation that precedes the lemma and the fact that up until the moment when the first valid improper query is asked, the decryption oracles in all three experiments behave in the same way and the experiments are, in fact, identical. ▮

Note that this argument does not refer in any way to computational power of $\mathcal{D}_1$; in particular, it holds for computationally unbounded $\mathcal{D}_1$.

$$
\begin{array}{|l|}
\hline
\mathsf{Expt}_3 : \\
\quad \text{Set up } PK, SK: \\
\qquad \sigma \leftarrow G^{\sigma}(1^k) \\
\qquad (pk, sk, h, d_1, d_2) \leftarrow G(\sigma) \\
\qquad (R, \rho, \gamma) \leftarrow S_1(\sigma) \\
\qquad PK = (pk, R), SK = (sk, \rho) \\
\quad (m_0, m_1, \alpha) \leftarrow A_1^{\tilde{\mathcal{D}}_1(d_1, \rho, \cdot)}(PK) \\
\quad \text{Set up challenge:} \\
\qquad c := \overline{E}(h, m_0) \qquad\qquad\qquad (*) \\
\qquad \pi := \mathcal{S}_2(c, R, \gamma) \\
\quad \texttt{return } A_2^{\tilde{\mathcal{D}}'_1(d_1, \rho, \cdot)}(c, \pi, \alpha) \\
\hline
\end{array}
$$

Given an A that can distinguish between Expt_3 and Expt_2 with nonnegligible probability, we can construct an adversary A' that has a nonnegligible advantage in distinguishing $\mathsf{Expt}^0(0)$ and $\mathsf{Expt}^1(0)$ in the definition of oblivious decryptors scheme. Namely, consider the following adversary $A' = (A'_1, A'_2)$:

$$
\begin{array}{|l|}
\hline
A'_1(\sigma) : \\
\ (R, \rho, \gamma) \leftarrow S_1(\sigma) \\
\ PK = (pk, R), SK = (sk, \rho) \\
\ (m_0, m_1, \alpha) \leftarrow A_1^{\tilde{\mathcal{D}}_1(d_1, \rho, \cdot)}(PK) \\
\ \texttt{return } (m_0, m_1, (\alpha, PK, \rho) \\
\hline
A'_2(c, (\alpha, PK, \rho) : \\
\ \pi := \mathcal{S}_2(c, R, \gamma) \\
\ \texttt{return } A_2^{\tilde{\mathcal{D}}'_1(d_1, \rho, \cdot)}(c, \pi, \alpha) \\
\hline
\end{array}
$$

A' implements $\tilde{\mathcal{D}}_1$ by checking the validity of the proof himself and then quering $\mathcal{D}_1$; for this A', we have $\Pr\left[\mathsf{Expt}^0(0)\right] - \Pr\left[\mathsf{Expt}^1(0)\right] = p_2 - p_3$.

Expt_4 :
  Set up PK, SK:
    $\sigma \leftarrow G^{\sigma}(1^k)$
    $(pk, sk, h, d_1, d_2) \leftarrow G(\sigma)$
    $(R, \rho, \gamma) \leftarrow S_1(\sigma)$
    $PK = (pk, R), SK = (sk, \rho)$
  $(m_0, m_1, \alpha) \leftarrow A_1^{\tilde{\mathcal{D}}_2(d_2,\rho,\cdot)}(PK)$ (*)
  Set up challenge:
    $c := \overline{E}(h, m_0)$
    $\pi := \mathcal{S}_2(c, R, \gamma)$
  `return` $A_2^{\tilde{\mathcal{D}}'_2(d_2,\rho,\cdot)}(c, \pi, \alpha)$ (*)

Now, the $\tilde{\mathcal{D}}_1$ oracle is replaced with $\tilde{\mathcal{D}}_2$ oracle, which which first runs $\mathcal{V}$ to check if it is given a valid ciphertext, rejects if this is not the case and calls $\mathcal{D}_2$ on the ciphertext otherwise. The argument in this case in based on simulation soundness and is identical to the one that shows that $|p_1 - p_2|$ is negligible.

Expt_5 :
  Set up PK, SK:
    $\sigma \leftarrow G^{\sigma}(1^k)$
    $(pk, sk, h, d_1, d_2) \leftarrow G(\sigma)$
    $(R, \rho, \gamma) \leftarrow S_1(\sigma)$
    $PK = (pk, R), SK = (sk, \rho)$
  $(m_0, m_1, \alpha) \leftarrow A_1^{\tilde{\mathcal{D}}_2(d_2,\rho,\cdot)}(PK)$
  Set up challenge:
    $c := \overline{E}(h, m_1)$ (*)
    $\pi := \mathcal{S}_2(c, R, \gamma)$
  `return` $A_2^{\tilde{\mathcal{D}}'_2(d_2,\rho,\cdot)}(c, \pi, \alpha)$

Finally, we replace $c := \overline{E}(h, m_0)$ with $c := \overline{E}(h, m_1)$. The indistinguishability in this case follows from the fact that $\mathsf{ODMExpt}^2(0)$ and $\mathsf{ODMExpt}^2(1)$ are indistinguishable; the reduction is similar to the previous case.

The hybrids Expt_6, Expt_7, Expt_8, Expt_9 can be obtained from Expt_3, Expt_2, Expt_1, Expt_0, respectively, by replacing $E(pk, m_0)$ with $E(pk, m_1)$ and $\overline{E}(h, m_0)$ with $\overline{E}(h, m_1)$; the arguments for the remaining cases are symmetric, so we are done.

6 Smooth Hash Proof Systems and ODM

Another example of oblivious decryptors model is provided by *smooth hash proof systems (HPS)* of [5].

Recall that a smooth HPS for a subset membership problem $\mathbf{M}$ is given by a collection of probability distributions $(I_l)_{l \geq 0}$ on *instance descriptions* of the form $\Lambda[X, L_0, W, R]$ together with an *instance sampling algorithm* $Samp_M$, which on input 1^l produces an instance Λ according to I_l and a *subset sampling algorithm* $Samp_L$, which takes as inputs 1^l and $\Lambda[X, L_0, W, R] \in I_l$ and produces a random $x \in L_0$ and a witness $w \in W$ for x, an associated *projective hash family* $\mathbf{H}$ described by a tuple $(\mathcal{H}, K, X, L_0, \Pi, S, \alpha)$, an algorithm $Samp_K$, which chooses k from K at random, an algorithm $Eval_\alpha$, which computes $\alpha(k) \in S$

given $k \in K$, a *private evaluation algorithm* f_s, which computes $h_k(x) \in \Pi$ given $k \in K$ and $x \in X$, and a *public evaluation algorithm* f_p, which computes $h_k(x) \in \Pi$ given $\alpha(k) \in S$, $x \in L_0$, and $w \in W$.

To show that a smooth HPS is a particular case of the oblivious decryptors model, we have to assume that there exists an efficient algorithm $Samp_X$ that samples X at random and also that for all $\Lambda = \Lambda[X, L, W, R]$ the relation R is an NP-relation; although this does not follow from the definition of a universal HPS, this assumption is rather natural and trivially holds for all examples described in [5].

We describe $G^\sigma, G, E, D, \overline{E}, \mathcal{D}_1, \mathcal{D}_2$, and L. In what follows, we suppress the dependence of $\mathbf{H}$ on Λ.

$G^\sigma(1^l)$(Public Parameter Generation) :
  $\Lambda[X, L, W, R] \leftarrow Samp_M(1^l)$
  $\sigma = (1^l, \Lambda[X, L, W, R])$

$G(\sigma)$(Key Generation) :
  $k \leftarrow Samp_K(\sigma)$
  $\alpha = Eval_\alpha(k)$
  $PK = (\alpha, \sigma), SK = (k, \sigma)$
  $h = SK, d_1 = SK, d_2 = PK$
  `return` (PK, SK, h, d_1, d_2)

$E(PK, m)$(Encryption) :
  $(x, w) \leftarrow Samp_L(\sigma)$
  `return` $(x, f_p(x, w, \alpha) \oplus m)$

$D(x, c, SK)$(Decryption) :
  `return` $f_s(x, k) \oplus c$

$\mathcal{D}_1(d_1, x, c)$(First Decryption Oracle) :
  `return` $f_s(x, k) \oplus c$

$\mathcal{D}_2(d_2, x, c)$(Second Decryption Oracle) :
  Guess w such that $(x, w) \in R$
  `return` $f_p(x, w, \alpha) \oplus c$

$\overline{E}(h, m_b)$(Invalid Ciphertext Sampling Algorithm) :
  $x \leftarrow Samp_X(\sigma)$
  `return` $(x, f_s(x, k) \oplus m_b)$

The language L is defined as the set of all pairs of the form $((1^l, \Lambda[X, L_0, W, R]), x)$, where $x \in L_0$.

Remark 6.1 In most examples of this scheme, the message space is a group rather than the set of binary strings of certain length; in this cases, $\oplus$ can be replaced by an appropriate function on the group, which is usually the group operation itself. The changes in the construction and proofs are straightforward.

Theorem 6.2 *The construction presented above satisfies the definition of oblivious decryptors encryption scheme.*

Proof: There are two indistinguishability conditions to be checked.

The difference between $\mathsf{Expt}^0(b)$ and $\mathsf{Expt}^1(b)$ in this case is that in one of them the ciphertext is $(x, f_p(x, w, \alpha) \oplus m_b)$ with x coming from $Samp_L$, while in the other one the ciphertext is $(x, f_s(x, k) \oplus m_b)$ with x coming from $Samp_X$. If we can distinguish between these two, we can break the assumption that M is a hard subset membership problem.

In the experiment Expt^2, the ciphertexts are $(x, f_s(x, k) \oplus m_0)$ and $(x, f_s(x, k) \oplus m_1)$, where x comes from $Samp_X$; we prove that these are statistically indistinguishable by showing that none of them can be distinguished from $(x, r \oplus m_0) \equiv_s (x, r \oplus m_1)$, where r is a random element of Π. Indeed, suppose otherwise and consider the following adversary $\mathcal{A}$ for the smoothness condition:

$$\boxed{\begin{array}{l} \mathcal{A}(PK, x, s): \\ \quad (m_0, m_1, \alpha) \leftarrow A_1^{\mathcal{D}_2(d_2,\cdot)}(PK) \\ \quad c := (x, s \oplus m_0) \\ \quad \texttt{return}\ A_2^{\mathcal{D}_2(d_2,\cdot)}(c, \alpha) \end{array}}$$

The string s is known to be either $f_s(x, k)$ or r, so the input to A_2 is $(x, f_s(x, k) \oplus m_0)$ or $(x, r \oplus m_0)$, respectively.

An adversary $\mathcal{A} \in P^{NP}$ can implement the decryption oracle $\mathcal{D}_2$: given (x, c), it uses its NP oracle to guess the witness w for x and computes $f_p(x, w, \alpha) \oplus c$. If $\mathcal{A}$ has a nonnegligible advantage in this experiment, we get a contradiction, since we assume that $f_s(x, k)$ is statistically indistinguishable from r, that is, even computationally unbounded adversary cannot distinguish between them with nonnegligible probability.

The case of $c := (x, s \oplus m_1)$ is handled similarly. ▮

Remark 6.3 Note that this construction does not use universal$_2$ projective hash functions; this is to be expected, as 2-universality can be viewed as a group-theoretic analogue of simulation-soundness.

7 Limitations on Universal Hash Proofs

We note that the definition of universal projective hashing given by [5] implies that these proof systems can only exist for languages that are in SZK.

Theorem 7.1 *Any language admitting* $1/2$*-universal HPS can be reduced to* ENTROPY DISTANCE*, a complete problem for SZK [15, 11].*

Proof: Consider the strongly universal HPS $\mathbf{P}$ for this language, which can be constructed from the $1/2$-universal HPS as described in [5].

Given an instance $x \in X$, $\Lambda = \Lambda[X, L_0, W, R] \in I_l$, $(\mathcal{H}, K, X, L_0, \Pi, S, \alpha) = \mathbf{H}(\Lambda)$ we produce two circuits: C_1 samples a random index $k \in K$ and outputs $(x, \alpha(k), h_k(x))$, while C_2 samples a random index $k \in K$ and outputs only $(x, \alpha(k))$.

Clearly, it follows from the definition of universal hashing that if $x \in L$, then $\alpha(k)$ fully determines $h_k(x)$ and the entropies of the distributions sampled by the two circuits above are identical.

On the other hand, if $x \notin L$, consider the corresponding $\epsilon(l)$-universal projective hash family $\mathbf{H}^*$, where ϵ is a negligible function. We have $H(x, \alpha^*(k^*), h^*_{k^*}(x))) = H(x, \alpha^*(k^*)) + H(h^*_{k^*}(x) \mid x, \alpha^*(k^*)) \geq H(x, \alpha^*(k^*)) + \log 1/\epsilon(l)$; furthermore, since $\mathbf{H}$ and $\mathbf{H}^*$ are δ-close for some negligible $\delta(l)$, the difference

$$|H(x, \alpha^*(k^*), h^*_{k^*}(x)) - H(x, \alpha(k), h_k(x))|$$

is negligible (this fact can be deduced from Fano's Inequality, see [2]), and hence $H(x, \alpha(k), h_k(x))$ and $H(x, \alpha(k))$ are far apart. ▮

Note that no problem in SZK can be NP-complete unless the polynomial-time hierarchy collapses [1, 8, 15].

Remark 7.2 Using similar methods, one can prove that if $H(\alpha(k))$ can be efficiently approximated, then the language in question can be reduced to ENTROPY APPROXIMATION and thus belongs to NISZK. As NISZK $\subseteq$ SZK, this result strenthens the previous theorem.

References

[1] W. AIELLO AND J. HASTAD, Statistical zero-knowledge languages can be recognized in two rounds. *Journal of Computer and System Sciences*, 42(3):327–345, June 1991.

[2] T. M. COVER AND J. A. THOMAS, *Elements of Information Theory.* Wiley Series in Telecommunications. John Wiley & Sons, Inc., 2nd edition, 1991.

[3] R. CRAMER AND V. SHOUP, A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. *Advances in Cryptology – Crypto* 98 *Proceedings*, Lecture Notes in Computer Science Vol. 1462, H. Krawczyk ed., Springer-Verlag, 1998.

[4] R. CRAMER AND V. SHOUP, Manuscript, June 2001.

[5] R. CRAMER AND V. SHOUP, Universal Hash Proofs and a Paradigm for Adaptive Chosen Ciphertext Secure Public-Key Encryption. IACR Cryptology E-Print Archive, October 2001.

[6] D. DOLEV, C. DWORK, AND M. NAOR, Non-Malleable Cryptography. *Proceedings of the* -45th *Annual Symposium on Theory of Computing*, ACM, 1923 and SIAM Journal on Computing, 2000.

[7] U. FEIGE, D. LAPIDOT, AND A. SHAMIR, Multiple non-interactive zero knowledge proofs based on a single random string (extended abstract). In *31st Annual Symposium on Foundations of Computer Science*, volume I, pages 308–317, St. Louis, Missouri, 22–24 October 1990. IEEE.

[8] L. FORTNOW, The complexity of perfect zero-knowledge. In Silvio Micali, editor, *Advances in Computing Research*, volume 5, pages 327–343. JAC Press, Inc., 1989.

[9] O. GOLDREICH, Foundations of cryptography. Class notes, Spring 1989, Technion University.

[10] O. GOLDREICH, A. SAHAI, AND S. VADHAN, Can statistical zero-knowledge be made non-interactive?, or On the relationship of SZK and NISZK. In *Advances in Cryptology—CRYPTO '99*, Lecture Notes in Computer Science, pages 467–484. Springer-Verlag, 1999, 15–19 August 1999.

[11] O. GOLDREICH AND S. VADHAN, Comparing entropies in statistical zero-knowledge with applications to the structure of SZK. In *Proceedings of the Fourteenth Annual IEEE Conference on Computational Complexity*, pages 54–73, Atlanta, GA, May 1999. IEEE Computer Society Press.

[12] M. NAOR AND M. YUNG, Public-key cryptosystems provably secure against chosen ciphertext attacks. *Proceedings of the* 22nd *Annual Symposium on Theory of Computing*, ACM, 1990.

[13] C. RACKOFF AND D. SIMON, Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. *Advances in Cryptology – Crypto* 91 *Proceedings*, Lecture Notes in Computer Science Vol. 576, J. Feigenbaum ed., Springer-Verlag, 1991.

[14] A. SAHAI, Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. *Proceedings of the* 40th *Symposium on Foundations of Computer Science*, IEEE, 1999

[15] A. SAHAI AND S. VADHAN A Complete Problem for Statistical Zero Knowledge. Preliminary version appeared in *Proceedings of the* 38th *Symposium on Foundations of Computer Science*, IEEE, 1997. Newer version may be obtained from authors' homepages.

[16] V. SHOUP, Why chosen ciphertext security matters, IBM Research Report RZ 3076, November, 1998.