Black-Box Complexity of Encryption and Commitment

Hoe Teck Wee

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2007-144
http://www.eecs.berkeley.edu/Pubs/TechRpts/2007/EECS-2007-144.html

December 6, 2007

Black-Box Complexity of Encryption and Commitment

by

Hoeteck Wee

B.S. (Massachusetts Institute of Technology) 2002

A dissertation submitted in partial satisfaction of the
requirements for the degree of

Doctor of Philosophy

in

Computer Science

in the

GRADUATE DIVISION

of the

UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:

Professor Luca Trevisan, Chair
Professor Christos Papadimitriou
Professor Elchanan Mossel

Fall 2007

Black-Box Complexity of Encryption and Commitment

Abstract

Black-Box Complexity of Encryption and Commitment

by

Hoeteck Wee

Doctor of Philosophy in Computer Science

University of California, Berkeley

Professor Luca Trevisan, Chair

We study the black-box complexity of non-malleable encryption and statistically hiding commitments. We present a black-box construction of a non-malleable encryption scheme from any semantically secure one, and a lower bound on the round complexity of a class of black-box constructions of statistically hiding commitments from one-way permutations.

Professor Luca Trevisan

Dissertation Committee Chair

to Berkeley, where I learnt to do research,
and New York, where I did most of the research herein.

Contents

Acknowledgments

My graduate school experience spanned a little over five years across five cities. I am extremely fortunate to have Luca Trevisan as an advisor and a friend throughout the entire experience, getting me started on research and sticking with me even as I abandoned extractors. Luca taught me fairly early on that good coffee, good food, and a good time are imperative to doing research, and he never faulted me when I picked fun over work. His insight and advice on navigating research, the academic community, and the world beyond helped me mature as a researcher, an academic and a young adult. I aspire to be like Luca when I grow up — to work hard and play hard in a big city and to go through life with uncharacteristic humor and lots of sleep.

The general consensus is that there are only two excellent places to study cryptography and I gave up one to go to Berkeley. I was very lucky to meet Salil Vadhan before leaving Boston and I am extremely grateful for the many illuminating and inspiring discussions we had over the years. Salil and Luca generously gave of their time to help me develop my own research agenda, to instill in me the creativity and confidence to be a good researcher, and to engage me in a relentless pursuit of rigor in my research along with clarity in writing and in talks.

When I started out in graduate school, I would settle for a high-level understanding while reading papers and built-in templates while designing slides. Talking to and working with Andrej changed both of that. I have since learnt to reproduce quite a few proofs from scratch; I also wish and do believe some of Andrej's impeccable taste in colors, typography and layout had rubbed off on me. His excellent penmanship, however, remains elusive.

I spent two lovely and very productive summers in New York, one at IBM T.J. Watson and another at Columbia, thanks to my mentors and hosts Ran Canetti, Rosario Gennaro, Tal Rabin, Tal Malkin and Rocco Servedio. Special thanks to Ran for getting me to think more like a cryptographer and a New Yorker! Apart from New York, I spent a summer at Microsoft Silicon Valley, a year at Tsinghua University in Beijing, and a semester at IPAM, all of which were wonderful learning experiences. I thank Cynthia Dwork, Andy Yao and my hosts at IBM and Columbia for their hospitality and for remaining such dedicated mentors since.

I wish to thank my co-authors for the collaboration, particularly Seung Geol Choi, Dana Dachman-Soled and Tal Malkin for the result in this thesis, as well as Daniel Jackson, Madhu Sudan and Silvio Micali for initiating my passage into research, theory and cryptography. In addition to the afore-mentioned, I have had many delightful and enjoyable discussions with Iftach Haitner, Rafael Pass, Alon Rosen, Guy Rothblum, David Woodruff, Vinod Vaikuntanathan, Emanuele Viola and Ke Yang on subjects technical and otherwise, and I look forward to many more.

I express my gratitude towards Christos Papadimitriou, Elchanan Mossel and Alistair Sinclair for serving on my thesis and quals committees, and for being so accommodating of my faults and my requests. I also thank Henry Lin and Guy for delivering a talk and my thesis in my absence and La Shana Polaris and Chang Su for making administration less of a nightmare.

I could not have made it this far if not for the amazing people I met along the way, who always regarded with unfailingly good humor my inexplicable idiosyncrasies ranging from an inaptitude for driving to a ban on broccoli and burrito. I refer to the occupants and affiliates of 587 Soda, FIT 1-280 and IPAM as well as all my friends from home, college and graduate school. Amongst these people, I would like to single out Andrej, Kamalika Chaudhuri, Iordanis Kerenidis, Kenji Obata, Ranjit Jhala, Kunal Talwar and David Ratajczak, who took on the various roles of little sister, big sibling, room-mate, care-takers of a man on crutches, and companions on numerous culinary expeditions, and also Robert Kessell, Ben Myers, Richard Tran and Kim Truong, who made sure I never got lost in America. I hope everyone will keep trying to make me eat green stuff.

Cafes have been an indispensable part of my life in graduate school, notably Brewed Awakening, Cafe Strada and S.I.T. Cafe. They are more than a place where coffee and ideas are brewed; they also keep me from brooding.

Finally, I thank my family for their unconditional support, even though I never tell them what exactly I do and where exactly I am going.

DEC 1, 2007

Chapter 1

Prelude

A typical afternoon[1] chat at Brewed Awakening usually entails a digression beyond computer science. Even digressions into politics or art need not be entirely devoid of technical content:

- Can Sotheby's prevent a bidder in an electronic "sealed-bid" auction from consistently bidding a dollar higher than all the previous bidders?
- Can the President of the United States convince its people that there are weapons of mass destruction in Iraq without revealing the location of these weapons or aerial shots of weapon facilities?

In this thesis, we examine cryptographic mechanisms for addressing both of these questions.

[1] Morning chats are atypical.

Chapter 2

Introduction

The early work on cryptography introduced many surprising feasibility results on secure computation based on very minimal assumptions on the hardness of certain computational problems [Y86, GMW87]. The natural goal thereafter is to improve the efficiency and security guarantees of these constructions while maintaining the same minimal assumptions. In this thesis, we study the complexity of two fundamental cryptographic primitives: commitment and public-key encryption.

Commitment schemes. A commitment scheme is the digital analogue of sending messages in a locked box; once the box reaches the receiver, the sender can no longer change its contents, and the contents are hidden from the receiver until a later point when the sender reveals the message by sending along the key. We choose to focus on *statistically hiding commitments*, wherein it is *computationally infeasible* for the sender to change the message and *statistically impossible* for the receiver to learn anything about the message until the reveal phase. Commitment schemes of this kind allow us to prove any NP statement in statistical zero knowledge [GMR89]: that is, a prover can convince a verifier of the validity of the statement in such a way that it is computationally infeasible to convince the verifier of a false statement, and statistically impossible for the verifier to learn any additional knowledge about the statement apart from its validity. The political analogy is quite apt here: a politician only needs to be convincing before and during his tenure, but may wish the data underlying his arguments to be completely withheld long after stepping down.

Public-key encryption schemes. A public-key encryption scheme is the digital analogue of a locked mailbox with a mail slot; the mail slot is exposed to the public, and anyone can drop off messages in the slot, but only the person who possesses the key can open the mailbox and read the message. In an electronic auction, a bidder can encrypt and submit her bid over a public channel using a public-key encryption scheme and be assured that her bid remains hidden from other bidders; this privacy guarantee is formalized by the notion of semantic security [GM84]. However, even with this privacy guarantee, it remains conceivable that another bidder can modify (maul) an encrypted bid to obtain an encryption of a higher bid, without learning what the original bid is. Indeed, many semantically secure encryption schemes are susceptible to such attacks. A *non-malleable encryption scheme* [DDN00] is one for which such attacks are ruled out: given an encryption of some message, it is infeasible to generate encryptions of a related message. When used in an auction, it guarantees that any bid must be "computationally independent" of the previous ones.

2.1 Cryptography from General Assumptions

Much of the modern work in foundations of cryptography rests on general cryptographic assumptions like the existence of one-way functions and trapdoor permutations. General assumptions provide an abstraction of the functionalities and hardness we exploit in specific assumptions such as hardness of factoring and discrete log without referring to any specific underlying algebraic structure. The expressive nature of general assumptions means that we could then derive constructions based on a large number of concrete assumptions of our choice, even ones that may not have been considered at the time of designing the protocols. Indeed, the use of general assumptions allows more theoretical research in cryptography (say, notions of security and general secure multiparty computation) to proceed independently and concurrently with more practical research in cryptanalysis, design of specific cryptographic hash functions and the search for new concrete assumptions, e.g. those based on elliptic curves and lattices. It should also be noted that the rich algebraic structure is precisely what makes specific assumptions like hardness of factoring and computing discrete log susceptible to non-trivial attacks such as the number sieve method, index calculus, and quantum algorithms; as such, the absence of rich structure

makes a general assumption arguably more plausible.

On the flip side, the general lack of structure in general assumptions also makes protocol design a lot more difficult. The vast majority of constructions in cryptography merely treat the general assumption as a black box, that is, they refer only to the input/output behavior of the underlying functionality and not the code computing that functionality (notable exceptions include [GMW87, FS89, DDN00, B01, AIK06]). For instance, a black-box construction based on secure public-key encryption would treat the algorithms computing key generation, encryption, and decryption as oracles. Motivated by the prevalence and importance of black-box constructions in cryptography, a rich and fruitful body of work initiated in [IR89] seeks to understand the power and limitations of black-box constructions in cryptography. These works may be broadly classified as follows (in roughly chronological order):

- black-box separations amongst cryptographic tasks and primitives, e.g. one-way functions, collision-resistant hash functions, public-key encryption, key agreement, oblivious transfer [IR89, R91, S98, GMR01, RTV04];
- lower bounds on efficiency of cryptographic constructions, notably query complexity of pseudorandom generators, signature and encryption schemes [KST99, GT00, GGKT05, LTW05];
- black-box constructions where the only previous constructions are non-black-box, yet the existence of a black-box construction is not ruled out [IKLP06, CHH+07, PW07, H08].

We note that in cases where previous constructions are non-black-box, the new black-box constructions typically yield more efficient protocols that are simpler to describe and to implement.

2.2 Contributions & Organization

We present new results on the complexity of statistically hiding commitments and non-malleable encryption with respect to black-box constructions, which in turn provide new insight into the power and limitations of black-box constructions in cryptography.

Statistically hiding commitments. We know that statistically hiding commitments imply one-way functions (c.f. [IL89]), and until very recently, the only reverse connection is a commitment scheme based on one-way permutations with a linear number of rounds [NOVY98]. In Chapter 3, we show that the round complexity of this construction is essentially optimal, by providing a lower bound for a restricted class of black-box constructions. The restriction is on the structure of the construction; in addition, we also require that the proof of security is black-box, that is, we can use an adversary breaking the commitment as an oracle to break the underlying one-way permutation. Indeed, all known constructions in cryptography have black-box proofs of security (except in the context of simulation e.g. for zero knowledge). The key insight lies in exploiting the black-box proof of security, which has been used in several follow-up works of Haitner et al. [HHRS07, HHS08]. Our result appeared in TCC '07 [W07].

Non-malleable encryption. It is easy to see that a non-malleable encryption scheme is also semantically secure. In Chapter 4, we establish an equivalence – we present a black-box construction of a non-malleable encryption scheme from any semantically secure one, building on the recent non-black-box construction of Pass et al. [PSV06]. Our construction departs from the oft-used paradigm of re-encrypting the same message with different keys and then proving consistency of encryptions. Instead, we exploit an encoding of the message based on low-degree polynomials. We hope this result will contribute towards the resolution of an important open problem in this field: whether there is a black-box construction of encryption schemes secure against adaptive chosen-ciphertext attack assuming the existence of some low-level primitive. This result is joint work with Seung Geol Choi, Dana Dachman-Soled and Tal Malkin and will appear at TCC '08 [CDMW08].

Chapter 3

Statistically Hiding Commitments

3.1 Introduction

A *zero-knowledge proof* is a protocol in which one party, the prover, convinces another party, the verifier, of the validity of an assertion while revealing no additional knowledge. Introduced by Goldwasser, Micali and Rackoff in the 1980s [GMR89], zero-knowledge proofs have played a central role in the design and study of cryptographic protocols. In these applications, it is important to construct constant-round zero-knowledge protocols for NP under minimal assumptions. In many cases, a computational zero-knowledge argument system suffices, and we know how to construct such protocols for NP under the (essentially) minimal assumption of one-way functions [BJY97, OW93]. On the other hand, there are cases wherein we need stronger guarantees, namely a computational zero-knowledge proof system, or a statistical zero-knowledge argument system.[1] Surprisingly, the main bottleneck to reducing the assumptions for known constructions of both constant-round computational zero-knowledge proof systems and statistical zero-knowledge argument systems [BCY91, GK96a] is statistically hiding commitments.[2]

We know how to construct constant-round statistically-hiding commitments from

[1] It is unlikely that every language in NP has a statistical zero-knowledge proof system [F89, AH91, BHZ87].

[2] It is not surprising that we need statistically hiding commitments for statistical zero-knowledge arguments; what is surprising is that the only known approach for constructing constant-round zero-knowledge proof systems [GK96a] requires statistically hiding commitments to guarantee soundness, because the verifier begins by committing to her challenges.

collision-resistant hash functions [DPP98, NY89] and from claw-free permutations [GK96a]. In 1992, Naor, Ostrovsky, Venkatesan and Yung [NOVY98] showed that one-way permutations are sufficient for statistically hiding commitments. This was very recently extended to one-way functions by Haitner and Reingold [HR07b]. Both works use the powerful tool of interactive hashing [OVY93], a 2-party protocol for choosing a small set of strings, with binding and hiding requirements similar to those in commitment schemes, and the round complexity of the protocols are at least linear in the security parameter. An intriguing open problem (posed in [NOVY98] and reiterated in [DHRS04, KS06, HR07a]) is whether some variant of interactive hashing could yield a constant-round statistically hiding commitment from one-way permutations. In fact, even a $n^{o(1)}$-round commitment would be interesting. The restriction to interactive hashing may seem limiting, but it is the only technique that we presently know of. Moreover, Ding, et al. [DHRS04] exhibited a constant-round interactive hashing protocol satisfying a weaker binding guarantee, which indicates that interactive hashing may not be the bottleneck.

3.1.1 Our contributions and techniques

We study a natural class of black-box constructions of statistically hiding commitments from one-way permutations that include several generalizations of the NOVY construction, and show that any such construction yields a commitment scheme with at least $\Omega(n/\log n)$ rounds. This matches the round complexity of a variant of the main NOVY construction ([KS06, HR07a]). Specifically, our lower bound holds for constructions in which the sender (in the commitment scheme) evaluates the one-way permutation only at the start of the commit phase, and does so on independent random inputs. The sender then uses the output values, her private input to the commitment scheme, and possibly additional randomness in the rest of the commit phase and does not use the inputs to the one-way permutation until the reveal phase.

We derive as a corollary, a $\Omega(n/\log n)$ lower bound on a computational form of interactive hashing presented in [NOV06, HR07a], based on an abstraction of the way interactive hashing is used in the NOVY construction and in the subsequent works of Haitner et al. [HHK+05, NOV06, HR07b]. The same abstraction also applies to the use of interactive hashing in the transformation of honest-verifier zero-knowledge arguments into cheating-

verifier zero-knowledge arguments [D93, OVY93]. The lower bound tells us that we need to avoid the standard notion of interactive hashing if we want round-efficient versions of these applications.

Our lower bound for statistically hiding commitments only holds for fully black-box constructions [RTV04], namely, we require not only that the construction treats the one-way permutation as a black-box, but also that the reduction in the proof of security uses black-box access to a cheating sender that breaks the binding property to invert the permutation with noticeable probability. At a high level, our lower bound follows the paradigm of Gennaro and Trevisan [GT00] for proving lower bounds on efficiency of black-box cryptographic constructions, which is in turn based on the Impagliazzo-Rudich framework [IR89] for separating cryptographic primitives. The proof techniques and ideas are otherwise largely inspired by lower bounds for black-box zero-knowledge from the work of Goldreich and Krawczyk [GK96b].

Roughly speaking, a fully black-box construction guarantees an efficient procedure that by interacting and rewinding the cheating sender, produces transcripts of the commit phase that are "consistent" with the input to the reduction. Using the repeated sampling technique from [IR89], we can ensure that the probability of seeing a consistent partial transcript is exponentially small in the length of the sender's last message. This means that the sender sends $O(\log n)$ bits in each round of protocol. On the other hand, the sender must send a total of $\Omega(n)$ bits in the protocol (so that there is a different transcript for every possible challenge for the one-way permutation), which means the protocol must have $\Omega(n/\log n)$ rounds. This simplified and slightly inaccurate sketch overlooks several technical difficulties.

3.1.2 Perspective

Notions and limitations of interactive hashing. The last few years have witnessed a lot of work on the use of interactive hashing protocols in cryptography with two main notions of security: computationally binding, and binding for static sets [NOV06]. The latter is used in building and studying oblivious transfer protocols in the bounded storage model and over noisy channels [CCM98, DHRS04, CS06], in constructing variants of statistically binding commitments [NV06], and in transforming honest-verifier zero-knowledge proofs

into cheating-verifier zero-knowledge proofs [D93, DGOW95, GSV98]. It was noted in [NOV06, CCM98] that the computational binding implies binding for static sets; our lower bound implies that the converse is not true. Specifically, the constant-round protocol of [DHRS04] does not satisfy the computational formulation (which answers an open problem in [DHRS04] in the negative).

Efficiency of cryptographic reductions. Previous work establishing lower bounds for efficiency of black-box cryptographic reductions has focused on the query complexity and randomness complexity of these reductions [KST99, GGKT05, LTW05, HK05] whereas our work focuses on round complexity. Upon closer inspection, our work is also qualitatively very different (apart from studying a different computational resource) as the lower bounds of [GGKT05, LTW05, HK05] apply also to weakly black-box constructions, in which the proof of security may exploit the code of the adversary (in a non-black-box manner). As mentioned earlier, our main result only rules out fully black-box reductions and uses fairly different techniques. We stress that all known reductions between cryptographic primitives - with the exception of the non-black-box techniques used in zero-knowledge and multi-party protocols, e.g. [B01], but including the non-black-box constructions in [AIK06] - do not exploit the code of the adversary in the proof of security. As such, ruling out fully black-box constructions is almost as meaningful as ruling weakly black-box constructions.

Information-theoretic analogues. Many black-box cryptographic constructions apart from interactive hashing-based commitments have an information-theoretic analogue which is easier to achieve, in that it does not have some kind of "simulateable" requirement, namely, an efficient procedure for simulating random transcripts with a certain outcome. This was articulated in [DGW95], using random selection as a case study. In [LTW05], a formal connection between hardness amplification and combinatorial hitters was used to derive lower bounds on query and randomness complexity of the former. While the resulting lower bounds on query complexity are tight, those for randomness complexity are far from the best-known constructions. The information-theoretic analogue for computational interactive hashing would be interactive hashing with binding for static sets, for which we cannot expect to prove a super-constant lower bound (again, due to the constant-round

protocol in [DHRS04]). Indeed, we exploit the "simulateable" requirement for our lower bound in a very essential way.

Implications for protocol design. One could view this work quite broadly as providing a simple informal criterion for reasoning about the round complexity of classes of fully-black-box constructions (of protocols with a "simulatable" requirement) and formal techniques towards establishing a lower bound. The former is especially useful for protocol design in identifying and ruling out inefficient constructions. We stress here that our lower bounds do not apply to the black-box constructions of commitments from various classes of one-way functions in the works of Haitner et al. [HHK+05, NOV06, HR07b], in two different ways. One is the use of one-way functions in [HHK+05] to implement coin-tossing and zero-knowledge proofs to transform commitments that are hiding against honest receivers into commitments that are hiding against arbitrary receivers. We note that our lower bound holds assuming merely hiding against honest receivers. The second is that the inputs to the one-way functions are used again in the commit phase. This is only needed to handle the lack of structure in general one-way functions. In particular, all the constructions are much simpler and requires fewer rounds when optimized for one-way permutations - they "collapse" to the NOVY construction. In short, the ways in which these constructions bypass our lower bounds do not provide much insight into how we may bypass the lower bounds for one-way permutations.

Additional related work. Fischlin [F02] showed that there is no black-box construction of 2-message statistically hiding from one-way permutations (or even trapdoor permutations). The result follows quite readily from Simon's oracle separating collision-resistant hash functions and one-way permutations [S98]. On the other hand, Harnik and Naor [HN06] gave a non-black-box construction of a 2-message statistically hiding commitment from one-way functions under a non-standard assumption on compressibility of NP instances. From what we understand, there is no strong evidence either supporting or refuting the assumption.

3.1.3 Subsequent work

In follow-up work to this one, Haitner et al. [HHRS07] extended our lower bound to any fully black-box construction of statistically hiding commitments from one-way permutations. They used the same cheating sender as ours for the lower bound, along with an elegant analysis (building on [S98]) that allows them to bypass the structural restrictions we needed in our lower bound. To implement the cheating sender, they introduced a sampling oracle which they used in additional follow-up work [HHS08] to obtain linear communication complexity lower bounds for private information retrieval, again for fully black-box constructions.

3.2 Preliminaries & Definitions

We use ppt to denote probabilistic polynomial time. The *round complexity* of a 2-party protocol is number of pairs of messages exchanged by both parties (in both directions). Unless otherwise stated, we use 1^n as the security parameter.

3.2.1 One-way permutations

Definition 1. *A function $f : \{0,1\}^* \to \{0,1\}^*$ is a $s(n)$-*secure one-way function *if f is computable in polynomial time and for every nonuniform ppt machine A,*

$$\Pr_{x \in \{0,1\}^n}[A(1^n, f(x)) \in f^{-1}(f(x))] < 1/s(n)$$

A function f is a one-way permutation *if for every n, f restricted to $\{0,1\}^n$ is a permutation, and for all polynomials $s(n)$ and all sufficiently large n, f is $s(n)$-secure.*

A random permutation π is exponentially one-way even if the adversary is given access to a π^{-1} oracle, as long as it cannot query π^{-1} on the challenge. Here, $\pi^{-1}_{\neq y}$ is an oracle that on input y', returns $\pi^{-1}(y')$ if $y' \neq y$, and $\perp$ otherwise.

Lemma 1 (implicit in [GT00]). *Fix $s(n) = 2^{n/5}$. For all sufficiently large n, there exists a permutation π on $\{0,1\}^n$ such that for all circuits A of size $s(n)$,*

$$\Pr_{y \in \{0,1\}^n}[A^{\pi, \pi^{-1}_{\neq y}}(y) = \pi^{-1}(y)] < \frac{1}{s(n)}$$

Moreover, the statement relativizes.

3.2.2 Statistically hiding commitments

We present the definition for bit commitment. To commit to multiple bits, we may simply run a bit commitment scheme in parallel.

Definition 2. *A* (bit) commitment scheme $(\mathcal{S}, \mathcal{R})$ *is an efficient two-party protocol consisting of two stages. Throughout, both parties receive the security parameter 1^n as input.*

> Commit. *The sender $\mathcal{S}$ has a private input $b \in \{0,1\}$, which she wishes to commit to the receiver $\mathcal{R}$, and a sequence of coin tosses σ. At the end of this stage, both parties receive as common output a commitment z.*
>
> Reveal. *Both parties receive as input a commitment z. $\mathcal{S}$ also receives the private input b and coin tosses σ for z. This stage is non-interactive: $\mathcal{S}$ sends a single message to $\mathcal{R}$, and $\mathcal{R}$ either outputs a bit and accepts or rejects.*

Definition 3. *A commitment scheme $(\mathcal{S}, \mathcal{R})$ is* perfectly hiding *if*

> Completeness. *If both parties are honest, then for any input bit $b \in \{0,1\}$ that $\mathcal{S}$ gets, $\mathcal{R}$ outputs b and accepts at the end of the decommit stage.*
>
> Statistically Hiding. *For every unbounded deterministic strategy $\mathcal{R}^*$, the distributions of the view of $\mathcal{R}^*$ in the commit stage while interacting with an honest $\mathcal{S}$ are identical for $b = 0$ and $b = 1$. If the distributions are statistically indistinguishable, we obtain a* statistically hiding *commitment.*
>
> Computationally Binding. *For every nonuniform ppt machine $\mathcal{S}^*$, $\mathcal{S}^*$ succeeds in the following game (breaks the commitment) with negligible probability:*

– *$\mathcal{S}^*$ interacts with an honest $\mathcal{R}$ and outputs a commitment z.*

– *$\mathcal{S}^*$ outputs two messages τ_0, τ_1 such that for both $b = 0$ and $b = 1$, $\mathcal{R}$ on input (z, τ_b) accepts and outputs b.*

3.3 Commitments from One-Way Permutations

In this section, we provide formal definitions of the various classes of constructions of commitments from one-way permutations we consider in this paper.

3.3.1 Fully black-box constructions

Definition 4. *A fully black-box construction of a statistically hiding commitment scheme from one-way permutations is a triplet of ppt oracle procedures $(\mathcal{S}, \mathcal{R}, M)$ for which there exists a polynomial T and a constant c satisfying the following properties:*

EFFICIENCY. *The running times of $\mathcal{S}, \mathcal{R}, M$ are bounded by T.*

FUNCTIONALITY. *For every family of permutations π, $(\mathcal{S}^\pi, \mathcal{R}^\pi)$ is a statistically hiding commitment scheme.*

SECURITY. *For every $\varepsilon = 1/\operatorname{poly}(n)$, for all sufficiently large n, every permutation $\pi : \{0,1\}^n \to \{0,1\}^n$ and every adversary $\mathcal{S}^*$, if $\mathcal{S}^*$ breaks $(\mathcal{S}^\pi, \mathcal{R}^\pi)$ with probability ε, then*

$$\Pr_{y \in \{0,1\}^n}[M^{\mathcal{S}^*,\pi}(y) = \pi^{-1}(y)] \geq \left(\frac{\varepsilon}{T}\right)^c$$

3.3.2 Interactive hashing

Interactive hashing is a 2-party protocol between a sender and a receiver, similar to a commitment scheme. The sender begins with a private input $y \in \{0,1\}^q$ and goal is for both parties to select a set of 2^k strings in $\{0,1\}^q$ (specified by a circuit $C : \{0,1\}^k \to \{0,1\}^q$) containing y. The hiding property stipulates that the receiver does not learn which of the 2^k strings equals y, and the binding property stipulates that the sender can "control" at most one of the 2^k strings. The computational formulation (introduced explicitly in [NOV06]

along with selecting many instead of merely 2 outputs) guarantees an efficient reduction from breaking the binding property to solving some computational problem on random instances.

Definition 5 ([NOV06]). *A* computational interactive hashing scheme (with multiple outputs) *is an efficient protocol* $(\mathcal{S}_{\mathsf{IH}}, \mathcal{R}_{\mathsf{IH}})$ *where both parties receive common inputs* $(1^q, 1^k)$*,* $\mathcal{S}_{\mathsf{IH}}$ *receives a private input* $y \in \{0,1\}^q$*, with the common output being a circuit* $C : \{0,1\}^k \to \{0,1\}^q$ *and the private output of* $\mathcal{S}_{\mathsf{IH}}$ *being a string* $z \in \{0,1\}^k$*. The protocol satisfies the following properties:*

> Correctness. *For all* $\mathcal{R}^*$ *and all* $y \in \{0,1\}^q$*, let* C, z *be the common and private output of* $\mathcal{S}_{\mathsf{IH}}$ *in the protocol* $(\mathcal{S}_{\mathsf{IH}}, R^*)(1^q, 1^k)$*. Then,* $C(z) = y$*.*
>
> Perfectly Hiding. *For all* $\mathcal{R}^*$*,* (V, Z) *is distributed identically to* (V, U_k)*, where* $V = \mathsf{view}_{\mathcal{R}^*}(\mathcal{S}_{\mathsf{IH}}(U_q,), \mathcal{R}^*)$*.*
>
> Computationally Binding. *There exists an oracle ppt machine* A *such that for every* $\mathcal{S}^*$ *and any relation* W*, letting* $C, ((x_0, z_0), (x_1, z_1)$ *be the common and private output of* $\mathcal{S}_{\mathsf{IH}}$ *in the protocol* $(\mathcal{S}_{\mathsf{IH}}, R^*)(1^q, 1^k)$*, if it holds that*
>
> $$\Pr[(x_0, C(z_0)) \in W \wedge (x_1, C(z_1)) \in W \wedge z_0 \neq z_1] > \varepsilon,$$
>
> *where the above probability is over the coin tosses of* $\mathcal{R}_{\mathsf{IH}}$ *and* $\mathcal{S}^*$*, then we have that*
>
> $$\Pr_{y \in \{0,1\}^q}[(A^{\mathcal{S}^*}(y, 1^q, 1^k, \varepsilon), y) \in W] > 2^{-k} \cdot (\varepsilon/q)^{O(1)}.$$

Nguyen et al. [NOV06] presented a protocol satisfying the above definition with $q-k$ rounds, obtained by ending the NOVY protocol $k-1$ rounds earlier. The protocol is very simple: the receiver chooses $q-k$ linearly independent vectors $v_1, \ldots, v_k$ over $\{0,1\}^q$. In round i, the receiver sends v_i and the sender responds with bit-wise dot product $v_i \cdot y$. We may reduce the round complexity by a factor of $O(\log q)$ by having the receiver send a pairwise independent hash function $h_i : \{0,1\}^q \to \{0,1\}^{O(\log q)}$ in round i and the sender responding with $h_i(y)$ [HR07a]. Note that the sender is deterministic, and the protocol is public-coin. Our lower bound shows that using a randomized sender or a private-coin protocol or q-wise

independent hash functions will not further improve the round complexity (beyond constant factors).

Returning to the above definition, note that it refers to general relations W that may not be polynomial-time computable, and it does not give A oracle access to the relation W, which strengthens the security guarantee of the [NOV06] protocol. Our lower bound holds even if A has oracle access to the relation W, which is a weaker guarantee and thus a stronger lower bound. We also note that we may use the techniques in [LTW05] to show that this weaker guarantee also implies binding for static sets, thereby strengthening an observation made in [NOV06].

Naor et al. [NOVY98] showed that any computational interactive hashing scheme $(\mathcal{S}_{\mathsf{IH}}, \mathcal{R}_{\mathsf{IH}})$ yields a fully black-box construction of a perfectly hiding commitment scheme $(\mathcal{S}, \mathcal{R})$ from any one-way permutation π with essentially the same round complexity.[3] The construction is as follows:

> COMMIT. To commit to a bit b, $\mathcal{S}$ chooses a random $\sigma \in \{0,1\}^n$, where n is the security parameter. Then, $\mathcal{S}$ and $\mathcal{R}$ run as a sub-protocol $(\mathcal{S}_{\mathsf{IH}}(\pi(\sigma), \mathcal{R}_{\mathsf{IH}})(1^n, 1^1)$, playing the roles $\mathcal{S}_{\mathsf{IH}}, \mathcal{R}_{\mathsf{IH}}$ respectively. Let C, z be the common and private outputs of $\mathcal{S}$ in the sub-protocol. $\mathcal{S}$ then sends $b' = b \oplus z$.
>
> DECOMMIT. $\mathcal{S}$ sends (b, σ). $\mathcal{R}$ accepts and outputs b if $C(b \oplus b') = \pi(\sigma)$, and rejects otherwise.

We stress that in the construction, $\mathcal{S}$ queries π exactly once, to compute $\pi(\sigma)$, and does not need σ again except for decommitment.

As noted in the introduction, Damgård [D93] showed how any computational interactive hashing scheme can be used to transform constant-round honest-verifier public-coin zero-knowledge arguments into cheating-verifier public-coin zero-knowledge arguments unconditionally. The transformation may also be made more efficient by exploiting interactive hashing with multiple outputs so that a single application of interactive hashing yields a cheating-verifier zero-knowledge argument with soundness to $1/\operatorname{poly}(n)$ (instead of $1/2$).

[3] More precisely, Naor et al. showed how to construct a perfectly hiding commitment scheme from any one-way permutation using the interactive hashing protocol in [OVY93]. Implicit in the proof of correctness and security is a proof that the [OVY93] protocol satisfies Definition 5 for $k = 1$.

3.3.3 Preimage-oblivious constructions

We describe the syntactic constraints on the class of fully black-box constructions for which we prove a lower bound. We consider constructions in which the sender evaluates the one-way permutation only at the start of the commit phase, and does so on independent random inputs. The sender then uses the values (and not the inputs to the permutation), its input bit and possibly additional randomness in the rest of the commit phase. To decommit, the sender sends its input bit and its random tape, including the inputs to the permutation. We allow the receiver to query the permutation at any point in the protocol.

More formally,

Definition 6. *A fully black-box construction $(\mathcal{S}, \mathcal{R}, M)$ of a statistically hiding commitments from one-way permutations is* preimage-oblivious *if there exists some interactive ppt machine $\mathcal{S}_{\mathsf{ob}}$ such that for any permutation π on $\{0,1\}^n$, to commit to a bit b with coin tosses σ, $\mathcal{S}$ parses $\sigma = (\mathbf{z}, \tilde{\sigma})$, where $\mathbf{z} = (z_1, \ldots, z_t) \in (\{0,1\}^n)^t$, and proceeds according to $\mathcal{S}_{\mathsf{ob}}(b, \sigma')$, where $\sigma' = (\mathbf{z}', \tilde{\sigma})$ and $\mathbf{z}' = \pi(\mathbf{z}) = (\pi(z_1), \ldots, \pi(z_t))$. In particular, $\mathcal{S}_{\mathsf{ob}}$ never queries π. To decommit, $\mathcal{S}$ sends a single message (b, σ).*

Clearly, the NOVY construction is preimage-oblivious; there, $t = 1$ and $\mathcal{S}_{\mathsf{ob}} = \mathcal{S}_{\mathsf{IH}}$ gets input $\pi(z_1)$, and $\tilde{\sigma}$ is the empty string since $\mathcal{S}_{\mathsf{IH}}$ is deterministic. Other candidates of preimage-oblivious constructions include variants of the NOVY construction in which we run n^2 copies of some variant of interactive hashing in parallel either on the same $t = 1$ input $\pi(z_1)$ or on $t = n^2$ independent inputs $\pi(z_1), \ldots, \pi(z_t)$, or a single copy of interactive hashing on the tn-bit string $\pi(z_1), \ldots, \pi(z_t)$.

On the other hand, the construction of statistically hiding commitments from one-way functions in [HR07b] is not preimage-oblivious. This is because the sender will query π at some point z_1 and send both $h_1(\pi(z_1))$ and $h_2(z_1)$ during the commit phase, for some hash functions h_1, h_2.

3.4 Main Result: Lower Bound for Commitments

Now, we state and prove our main result:

Theorem 1. *Any preimage-oblivious fully black-box construction of a statistically hiding commitment scheme from one-way permutations yields a commitment scheme with* $\Omega(\frac{n}{\log n})$ *rounds. This holds even if the hiding property for commitment scheme only holds for the honest receiver. More generally, if we assume that permutation is* s*-secure one-way, then we have an* $\Omega(\frac{n}{\log s})$ *lower bound.*

Our lower bound is tight:

Theorem 2 ([NOVY98, KS06, HR07a]). *There is a preimage-oblivious fully black-box construction of a perfectly hiding commitment scheme from* s*-secure one-way permutations with* $O(\frac{n}{\log s})$ *rounds.*

3.4.1 Proof intuition

First, we point out at a high level how we exploit the fact that the construction is fully black-box. We use as the one-way permutation the one guaranteed by Lemma 1, which remains one-way even under a "chosen challenge" attack. This means that in order for the reduction M to successfully invert a challenge y, it must get a cheating sender $\mathcal{S}^*$ to invert π on y itself. However, M is only given black-box access to $\mathcal{S}^*$, so it is limited to sending $\mathcal{S}^*$ different inputs and possibly rewinding $\mathcal{S}^*$.

For concreteness, consider the NOVY construction of commitment schemes from one-way permutation using computational interactive hashing as a subprotocol. When trying to invert a challenge y, the reduction M tries to get the sender to generate a commitment that is consistent with her input to interactive hashing protocol being y (otherwise, the decommitments will not help to invert y). At each round of commit phase, the honest $\mathcal{S}_{\mathsf{IH}}$ reveals some information about her input $\pi(\sigma)$. At the end of the commit phase, she should have revealed $n-1$ bits of information about her input (since we're using interactive hashing to choose 2 strings). We claim, at each round, she can only reveal $O(\log n)$ bits of information about her input, which yields a $\Omega(n/\log n)$ lower bound on the number of rounds. Suppose there is some round where $\mathcal{S}_{\mathsf{IH}}$ reveals $\omega(\log n)$ bits of information. This means that there are $n^{\omega(1)}$ inputs to the interactive hashing protocol that are consistent with the partial transcript. Consider a cheating sender that at each round samples a random input y' that is consistent with the partial transcript and responds

as though her input to the interactive hashing protocol is y', then the probability that the reduction observes a transcript that is consistent with y is negligible. It is important that $\mathcal{S}_{\mathsf{IH}}$ does not query π, so that we may sample consistent partial transcripts using a PSPACE oracle. If $\mathcal{S}_{\mathsf{IH}}$ is deterministic, it is straight-forward to quantify "information" about the sender's input and turn this outline into a proof.

For general preimage-oblivious constructions, we construct the cheating sender in essentially the same way: at each round (for both the commit and reveal phases), the sender samples a random (b, σ') that is consistent with the partial transcript and responds as though her input to $\mathcal{S}_{\mathsf{ob}}$ is (b, σ') (where $\sigma' = (\mathbf{z}', \tilde{\sigma})$). The main technical difficulty in the analysis is in quantifying "information" about the sender's input. Indeed, how much information a message reveals about $\mathbf{z}$ depends on both b and $\tilde{\sigma}$. Also, for a fixed partial transcript, the set (and number) of $\mathbf{z}'$'s that are consistent with the given transcript may vary with different choices of $b, \tilde{\sigma}$.

3.4.2 Proof of Theorem 1

We may assume that the commitment scheme $(\mathcal{S}, \mathcal{R})$ runs in r rounds, with $\mathcal{R}$ going first. Let T, c be the polynomial and constant guaranteed by the fully black-box reduction. We will show that $r \gtrsim \frac{n - \log t}{8c \log T} = \Omega(\frac{n}{\log n})$. Suppose otherwise, and take π to be the permutation guaranteed by Lemma 1.

Conventions regarding M. Recall that the reduction M has oracle access to a sender $\mathcal{S}^*$ with which it inverts the permutation π. It can query $\mathcal{S}^*$ on sequences of messages of the form $\mathbf{q}_i = (q_1, \ldots, q_i)$ corresponding to the first i messages from $\mathcal{R}$ in the commit phase, or a message of the form $(\mathbf{q}_r, \texttt{decommit})$, requesting for a decommit to a previous commitment. M runs for at most T steps, and therefore makes at most T queries to $\mathcal{S}^*$. In addition, we may adopt WLOG the following simplifying assumptions on M by modifying M appropriately (as is the case with lower bounds for black-box zero-knowledge [GK96b]):

1. It never asks the same query twice.

2. If M queries the oracle with $\mathbf{q}_i$, it has queried the oracle with all proper prefixes of $\mathbf{q}_i$ (namely all sequences of the form $(q_1, \ldots, q_j)$ for $j \leq i$.)

Notations. We introduce some notations:

- $\mathcal{S}_{\mathsf{ob}}(b, \sigma', \mathbf{q}_i)$ denotes the $\mathcal{S}_{\mathsf{ob}}$'s response with input b, σ' and and the first i messages from $\mathcal{R}$ being $\mathbf{q}_i$.
- Given a partial transcript $(\mathbf{q}_i, \mathbf{a}_i) = (q_1, \ldots, q_i, a_1, \ldots, a_i)$ and $y \in \{0,1\}^n$, $\mathrm{Con}(\mathbf{q}_i, \mathbf{a}_i)$ is the set of inputs (b, σ') to $\mathcal{S}_{\mathsf{ob}}$ that would yield the transcript $(\mathbf{q}_i, \mathbf{a}_i)$; formally,

$$\mathrm{Con}(\mathbf{q}_i, \mathbf{a}_i) = \{(b, \sigma') \mid \mathcal{S}_{\mathsf{ob}}(b, \sigma', q_1, \ldots, q_j) = a_j, \forall j = 1, 2, \ldots, i\}$$

  and

$$\mathrm{Con}_y(\mathbf{q}_i, \mathbf{a}_i) = \{(b, \mathbf{z}', \tilde{\sigma}) \in \mathrm{Con}(\mathbf{q}_i, \mathbf{a}_i) \mid \exists j : z'_j = y\}$$

  In particular, $|\mathrm{Con}_y(\epsilon)|/|\mathrm{Con}(\epsilon)| = 1 - (1 - 2^{-n})^t \leq t2^{-n}$, where ϵ is the empty string (transcript).

Sender strategy $\mathcal{S}^*$. Consider the following sender strategy $\mathcal{S}^*$:

- Upon receiving a query of the form $(\mathbf{q}_{i-1}, q_i)$, look up previous replies $\mathbf{a}_{i-1}$. (For $i = 1$, $(\mathbf{q}_{i-1}, \mathbf{a}_{i-1}) = \epsilon$.) Sample uniformly at random[4] (b, σ') from the set $\mathrm{Con}(\mathbf{q}_{i-1}, \mathbf{a}_{i-1})$, and respond with $a_i = \mathcal{S}_{\mathsf{ob}}(b, \sigma', \mathbf{q}_i)$.
- Upon receiving a query of the form $(\mathbf{q}_r, \texttt{decommit})$, look up previous replies $\mathbf{a}_r$. Sample uniformly and independently at random $(b_0, \mathbf{z}_0, \tilde{\sigma}_0), (b_1, \mathbf{z}_1, \tilde{\sigma}_1)$ from the set $\mathrm{Con}(\mathbf{q}_r, \mathbf{a}_r)$, and send $(b_0, \pi^{-1}(\mathbf{z}_0), \tilde{\sigma}_0), (b_1, \pi^{-1}(\mathbf{z}_1), \tilde{\sigma}_1)$.

Note that in an interaction with an honest receiver $\mathcal{R}$, $\mathcal{S}^*$ breaks the commitment with probability $1/2 - \mathrm{neg}(n) > 1/4$. This is because the hiding property of the commitment scheme guarantees that a random decommitment is almost equally likely to be a 0 and a 1. Hence,

$$\Pr_{y \in \{0,1\}^n}\left[M^{\mathcal{S}^*, \pi}(y) = \pi^{-1}(y)\right] > \left(\frac{1}{4T}\right)^c$$

Analysis. Note that a PSPACE oracle suffices for simulating $\mathcal{S}^*$ in the commit phase, whereas a PSPACE oracle and a π^{-1} oracle suffice in the reveal phase. Fix an input y

[4] $\mathcal{S}^*$ can be made stateless by using a rT-wise independent family of hash functions, namely apply a hash function to the queries and use the output as randomness for uniform sampling [GK96b].

to M. We want to show that with high probability, we may efficiently simulate the the computation $M^{S^*,\pi}(y)$ given oracle access to $\mathsf{PSPACE}, \pi, \pi^{-1}_{\neq y}$.

We say that a partial transcript $(\mathbf{q}_i, \mathbf{a}_i)$ is *heavy* if

$$\frac{|\mathrm{Con}_y(\mathbf{q}_i, \mathbf{a}_i)|}{|\mathrm{Con}(\mathbf{q}_i, \mathbf{a}_i)|} > \gamma^{r+1-i}, \quad \text{where } \gamma = \left(\frac{t}{2^n}\right)^{\frac{1}{r+1}};$$

otherwise, we say that $(\mathbf{q}_i, \mathbf{a}_i)$ is *light*. In particular, ϵ is light, since $\frac{|\mathrm{Con}_y(\epsilon)|}{|\mathrm{Con}(\epsilon)|} \leq \gamma^{r+1}$. Informally, the quantity $\frac{|\mathrm{Con}_y(\cdot)|}{|\mathrm{Con}(\cdot)|}$ applied to a transcript $(\mathbf{q}_i, \mathbf{a}_i)$ is the density of "favorable" outcomes for the reduction M, wherein an outcome is favorable if in the decommitment, $\mathcal{S}^*$ inverts π on y. We want to show that with high probability, every transcript generated by $\mathcal{S}^*$ (in its interaction with M) is light, that is, the density of favorable outcomes is low.

Consider the queries M makes to $\mathcal{S}^*$:

– A commit phase query of the form $\mathbf{q}_i = (\mathbf{q}_{i-1}, q_i)$. Let $\mathbf{a}_{i-1}$ be $\mathcal{S}^*$'s answers to the prefixes. Observe that

$$\begin{aligned}
\frac{|\mathrm{Con}_y(\mathbf{q}_{i-1}, \mathbf{a}_{i-1})|}{|\mathrm{Con}(\mathbf{q}_{i-1}, \mathbf{a}_{i-1})|} &= \sum_{a_i} \frac{|\mathrm{Con}(\mathbf{q}_i, \mathbf{a}_{i-1}, a_i)|}{|\mathrm{Con}(\mathbf{q}_{i-1}, \mathbf{a}_{i-1})|} \cdot \frac{|\mathrm{Con}_y(\mathbf{q}_i, \mathbf{a}_{i-1}, a_i)|}{|\mathrm{Con}(\mathbf{q}_i, \mathbf{a}_{i-1}, a_i)|} \\
&= \sum_{a_i} \Pr[S^*(\mathbf{q}_i) = a_i] \cdot \frac{|\mathrm{Con}_y(\mathbf{q}_i, \mathbf{a}_{i-1}, a_i|}{|\mathrm{Con}(\mathbf{q}_i, \mathbf{a}_{i-1}, a_i)|} \\
&> \Pr[S^*(\mathbf{q}_i) \to a_i; (\mathbf{q}_i, \mathbf{a}_{i-1}, a_i) \text{ is heavy}] \cdot \gamma^{r+1-i}
\end{aligned}$$

This implies

$$\Pr[S^*(\mathbf{q}_i) \to a_i; (\mathbf{q}_i, \mathbf{a}_{i-1}, a_i) \text{ is heavy} \mid (\mathbf{q}_{i-1}, \mathbf{a}_{i-1}) \text{ is light}] < \gamma$$

– A reveal phase query of the form $(\mathbf{q}_r, \texttt{decommit})$. Let $\mathbf{a}_r$ be $\mathcal{S}^*$'s answers to $\mathbf{q}_r$. If $(\mathbf{q}_r, \mathbf{a}_r)$ is light, that is, $\frac{|\mathrm{Con}_y(\mathbf{q}_r, \mathbf{a}_r)|}{|\mathrm{Con}(\mathbf{q}_r, \mathbf{a}_r)|} \leq \gamma$, then with probability $1 - 2\gamma$, we can generate two independent random decommitments without inverting π on y.

Applying a union bound over that rT commit phase queries that M makes to $\mathcal{S}^*$, we have: with probability at least $1 - rT\gamma$, in every reveal phase query $(\mathbf{q}_r, \texttt{decommit})$ that M makes to $\mathcal{S}^*$, the transcript $(\mathbf{q}_r, \mathbf{a}_r)$ is light. Taking another union bound, we deduce that with probability $1 - (r+2)T\gamma$, we may efficiently simulate $M^{S^*,\pi}$ on input y with oracle access

to $\mathsf{PSPACE}, \pi, \pi_{\neq y}^{-1}$. Hence, there is an oracle ppt machine $\tilde{M}$ running in time $\mathrm{poly}(T, n)$ such that

$$\Pr_{y \in \{0,1\}^n}\left[\tilde{M}^{\mathsf{PSPACE},\pi,\pi_{\neq y}^{-1}}(y) = \pi^{-1}(y)\ \right] > \left(\frac{1}{4T}\right)^c - (r+2)T\gamma > \frac{1}{2}\left(\frac{1}{4T}\right)^c$$

a contradiction to π being one-way. $\square$

3.4.3 Lower bounds for interactive hashing

Using the connection between commitment schemes and computational interactive hashing described in Section 3.3.2, we derive a tight lower bound for the latter [NOV06, HR07a]:

Theorem 3. *Any computational interactive hashing scheme on common input* $(1^n, 1^k)$ *has* $\Omega(\frac{n}{\log n})$ *rounds, for* $k = o(1)$.

Chapter 4

Non-Malleable Encryption

4.1 Introduction

The most basic security guarantee we require of a public key encryption scheme is that of semantic security [GM84]: it is infeasible to learn anything about the plaintext from the ciphertext. In many cryptographic applications such as auctions, we would like an encryption scheme that satisfies the stronger guarantee of non-malleability [DDN00], namely that given some ciphertext c, it is also infeasible to generate ciphertexts of some message that is related to the decryption of c. Motivated by the importance of non-malleability, Pass, Shelat and Vaikuntanathan raised the following question [PSV06]:

> It is possible to *immunize* any semantically secure encryption scheme against malleability attacks?

Pass et al. gave a beautiful construction of a non-malleable encryption scheme from any semantically secure one (building on [DDN00]), thereby addressing the question in the affirmative. However, the PSV construction – as with previous constructions achieving non-malleability from general assumptions [DDN00, S99, L06] – suffers from the curse of inefficiency arising from the use of general NP-reductions. In this work, we show that we can in fact immunize any semantically secure encryption schemes against malleability attacks without paying the price of general NP-reductions:

Main theorem (informal) There exists a (fully) black-box construction of a non-malleable encryption scheme from any semantically secure one.

That is, we provide a wrapper program (from programming language lingo) that given any subroutines for computing a semantically secure encryption scheme, computes a non-malleable encryption scheme, with a multiplicative overhead in the running time that is quasi-linear in the security parameter. Before providing further details, let us first provide some background and context for our result.

4.1.1 Relationships amongst cryptographic primitives

Recent work on understanding the power and limitations of black-box constructions in cryptography turned to tasks for which the only constructions we have are non-black-box, yet the existence of a black-box construction is not ruled out. Two notable examples are general secure multi-party computation against a dishonest majority and encryption schemes secure against adaptive chosen-ciphertext (CCA2) attacks[1] (c.f. [GMW87, DDN00]).

The general question of whether we can securely realize these tasks via black-box access to a general primitive is not merely only of theoretical interest. A practical reason is related to efficiency, as non-black-box constructions tend to be less efficient due to the use of general NP reductions to order to prove statements in zero knowledge; this impacts both computational complexity as well as communication complexity (which we interpret broadly to mean message lengths for protocols and key size and ciphertext size for encryption schemes). Moreover, if resolved in the affirmative, we expect the solution to provide new insights and techniques for circumventing the use of NP reductions and zero knowledge in the known constructions.

Indeed, Ishai et al. [IKLP06] recently provided an affirmative answer for secure multi-party computation by exhibiting black-box constructions from some low-level primitive. Their techniques have since been used to yield secure multi-party computation via black-box access to an oblivious transfer protocol for semi-honest parties, which is

[1] These are encryption schemes that remain semantically secure even under a CCA2 attack, wherein the adversary is allowed to query the decryption oracle except on the given challenge. A CCA1 attack is one wherein the adversary is allowed to query the decryption oracle before (but not after) seeing the challenge.

complete (and thus necessary) for secure multi-party computation [H08]. The following problem remains open:

> Is it possible to realize CCA2-secure encryption via black-box access to a low-level primitive, e.g. enhanced trapdoor permutations or homomorphic encryption schemes?

Previous work addressing this question is limited to non-black-box constructions of CCA2-secure encryption from enhanced trapdoor permutations [DDN00, S99, L06]; nothing is known assuming homomorphic encryption schemes. In work concurrent with ours, Peikert and Waters [PW07] made substantial progress towards the open problem – they constructed CCA2-secure encryption schemes via black-box access to a new primitive they introduced called lossy trapdoor functions, and in addition, gave constructions of this primitive from number-theoretic and worst-case lattice assumptions. Unfortunately, they do not provide a black-box construction of CCA2-secure encryption from enhanced trapdoor permutations.

Our work may also be viewed as a step towards closing this remaining gap (and a small step in the more general research agenda of understanding the power of black-box constructions). Specifically, the security guarantee provided by non-malleability lies between semantic security and CCA2 security, and we show how to derive non-malleability in a black-box manner from the minimal assumption possible, i.e., semantic security. In the process, we show how to enforce consistency of ciphertexts in a black-box manner. This issue arises in black-box constructions of both CCA2-secure and non-malleable encryptions. However, our consistency checks only satisfy a weaker notion of non-adaptive soundness, which is sufficient for non-malleability but not for CCA2-security (c.f. [PSV06]). As a special case of our result, we obtain a black-box construction of non-malleable encryptions from any (poly-to-1) trapdoor function. Our results are incomparable with those of Peikert and Waters since we start from weaker assumptions but derive a weaker security guarantee.

Related positive results. A different line of work focuses on (very) efficient constructions of CCA2-secure encryptions under specific number-theoretic assumptions [CS98, CS02, CHK04]. Apart from those based on identity-based encryption, these constructions together with previous ones based on general assumptions can be described under the following

framework (c.f. [BFM88, NY90, RS91, ES02]). Start with some cryptographic hardness assumption that allows us to build a semantically secure encryption scheme, and then prove/verify that several ciphertexts satisfy certain relations in one of two ways:

- exploiting algebraic relations from the underlying assumption to deduce additional structure in the encryption scheme (e.g. homomorphic, reusing randomness) [CS98, CS02];
- apply a general NP reduction to prove in non-interactive zero knowledge (NIZK) statements that relate to the primitive [DDN00, S99, L06].

None of the previous approaches seems to yield black-box constructions under general assumptions. Indeed, our work (also [PW07]) does not use the above framework.

4.1.2 Our results

As mentioned earlier, we exhibit a black-box construction of a non-malleable encryption scheme from any semantically secure one, the main novelty being that our construction is black-box. While this is interesting in and of itself, our construction also compares favorably with previous work in several regards:

- *Improved parameters.* We improve on the computational complexity of previous constructions based on general assumptions. In particular, we do not have to do an NP-reduction in either encryption or decryption, although we do have to pay the price of the running time of Berlekamp-Welch for decryption. The running time incurs a multiplicative overhead that is quasi-linear in the security parameter, over the running time of the underlying CPA secure scheme. Moreover, the sizes of public keys and ciphertext are independent of the computational complexity of the underlying scheme.

- *Conceptual simplicity/clarity.* Our scheme (and the analysis) is arguably much simpler than many of the previous constructions, and like [PSV06], entirely self-contained (apart from the Berlekamp-Welch algorithm). We do not need to appeal to notions of zero-knowledge, nor do we touch upon subtle technicalities like adaptive vs non-adaptive NIZK. Our construction may be covered in an introductory graduate course on cryptography without requiring zero knowledge as a pre-requisite.

– *Ease of implementation.* Our scheme is easy to describe and can be easily implemented in a modular fashion.

We may also derive from our construction additional positive and negative results.

Bounded CCA2 non-malleability. Cramer et al. [CHH+07] introduced the bounded CCA2 attack, a relaxation of the CCA2 attack wherein the adversary is only allowed make an a-priori bounded number of queries q to the decryption oracle, where q is fixed prior to choosing the parameters of the encryption scheme. In addition, starting from any semantically secure encryption, they obtained[2]:

– an encryption scheme that is semantically secure under a bounded-CCA2 attack via a black-box construction, wherein the size of the public key and ciphertext are quadratic in q; and

– an encryption scheme that is non-malleable under a bounded-CCA2 attack via a non-black-box construction, wherein the size of the public key and ciphertext are linear in q.

Combining their approach for the latter construction with our main result, we obtain an encryption scheme that is non-malleable under a bounded-CCA2 attack via a black-box construction, wherein the size of the public key and ciphertext are linear in q.

Separation between CCA2 security and non-malleability. Our main construction has the additional property that the decryption algorithm does not query the encryption functionality of the underlying scheme. Gertner, Malkin and Myers [GMM07] referred to such constructions as shielding and they showed that there is no shielding black-box construction of CCA1-secure encryption schemes from semantically secure encryption. Combined with the fact that any shielding construction when composed with our construction is again shielding, this immediately yields the following:

> **Corollary (informal)** There exists no shielding black-box construction of CCA1-secure encryption schemes from non-malleable encryption schemes.

[2]While semantic security and non-malleability are equivalent under a CCA2 attack [DDN00], they are not equivalent under a bounded-CCA2 attack, as shown in [CHH+07].

Note that a CCA2-secure encryption scheme is trivially also CCA1-secure, so this also implies a separation between non-malleability and CCA2-security for shielding black-box constructions.

Our techniques. At a high level, we follow the cut-and-choose approach for consistency checks from [PSV06], wherein the randomness used for cut-and-choose is specified in the secret key. A crucial component of our construction is a message encoding scheme with certain locally testable and self-correcting properties, based on the fact that low-degree polynomials are simultaneously good error-correcting codes and a secret-sharing scheme; this has been exploited in the early work on secure multi-party computation with malicious adversaries [BGW88]. We think this technique may be useful in eliminating general NP-reductions in other constructions in cryptography (outside of public-key encryption).

Towards CCA2 Security? The main obstacle towards achieving full CCA2 security from either semantically secure encryptions or enhanced trapdoor permutations using our approach (and also the [PSV06] approach) lies in guaranteeing soundness of the consistency checks against an adversary that can adaptively determine its queries depending on the outcome of previous consistency checks. It seems conceivable that using a non-shielding construction that uses re-encryption may help overcome this obstacle.

4.1.3 Overview of our construction

Recall the DDN [DDN00] and PSV [PSV06] constructions: to encrypt a message, one (a) generates k encryptions of the same message under independent keys, (b) gives a non-interactive zero-knowledge proof that all resulting ciphertexts are encryptions of the same message, and (c) signs the entire bundle with a one-time signature. It is in step (b) that we use a general NP-reduction, which in return makes the construction non-black-box. In the proof of security, we exploit that fact that for a well-formed ciphertext, we can recover the message if we know the secret key for any of the k encryptions.

How do we guarantee that a tuple of k ciphertexts are encryptions of the same plaintext without using a zero-knowledge proof and without revealing any information about the underlying plaintext? Naively, one would like to use a cut-and-choose approach (as has

been previously used in [LP07] to eliminate zero-knowledge proofs in the context of secure two-party computation), namely decrypt and verify that some constant fraction, say $k/2$ of the ciphertexts are indeed consistent. There are two issues with this approach:

- First, if only a constant number of ciphertexts are inconsistent, then we are unlikely to detect the inconsistency. To circumvent this problem, we could decrypt by outputting the majority of the remaining $k/2$ ciphertexts.
- The second issue is more fundamental: decrypting any of the ciphertexts will immediately reveal the underlying message, whereas it is crucial that we can enforce consistency while learning nothing about the underlying message.

We circumvent both issues by using a more sophisticated encoding of the message m based on low-degree polynomials instead of merely making k copies of the message as in the above schemes. Specifically, we pick a random degree k polynomial p such that $p(0) = m$ and we construct a $k \times 10k$ matrix such that the i'th column of the matrix comprises entirely of the value $p(i)$. To verify consistency, we will decrypt a random subset of k columns, and check that all the entries in each of these columns are the same.

- The issue that only a tiny number of ciphertexts are inconsistent is handled using the error-correcting properties of low-degree polynomials; specifically, each row of a valid encoding is a codeword for the Reed-Solomon code (and we output $\perp$ if it's far from any codeword).
- Low-degree polynomials are also good secret-sharing schemes, and learning a random subset of k columns in a valid encoding reveals nothing about the underlying message m. Encoding m using a secret-sharing scheme appears in the earlier work of Cramer et al. [CHH+07], but they do not consider redundancy or error-correction.

As before, we encrypt all the entries of the matrix using independent keys and then sign the entire bundle with a one-time signature. It is important that the encoding also provides a robustness guarantee similar to that of repeating the message k times: we are able to recover the message for a valid encryption if we can decrypt *any* row in the matrix. Indeed, this is essentially our entire scheme with two technical caveats:

– As with previous schemes, we will associate one pair of public/secret key pairs with each entry of the matrix, and we will select the public key for encryption based on the verification key of the one-time signature scheme.

– To enforce consistency, we will need a codeword check in addition to the column check outlined above. The reason for this is fairly subtle and we will highlight the issue in the formal exposition of our construction.

Decreasing ciphertext size. To encrypt an n-bit message with security parameter k, our construction yields $O(k^2)$ encryptions of n-bit messages in the underlying scheme. It is easy to see that this may be reduced to $O(k \log^2 k)$ encryptions by reducing the number of columns to $O(\log^2 k)$.

4.2 Preliminaries & Definitions

Notation. We adopt the notation used in [PSV06]. We use $[n]$ to denote $\{1, 2, \ldots, n\}$. Again, we use ppt to denote probabilistic polynomial time. Computational indistinguishability between two distributions A and B is denoted by $A \stackrel{c}{\approx} B$ and statistical indistinguishability by $A \stackrel{s}{\approx} B$.

4.2.1 Semantically secure encryption

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

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

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

$\mathsf{IND}_b(\Pi, A, k)$:

$(\mathrm{PK}, \mathrm{SK}) \leftarrow \mathsf{Gen}(1^k)$

$(m_0, m_1, \mathrm{STATE}_A) \leftarrow A_1(\mathrm{PK})$ *s.t.* $|m_0| = |m_1|$

$y \leftarrow \mathsf{Enc}_{\mathrm{PK}}(m_b)$

$D \leftarrow A_2(y, \mathrm{STATE}_A)$

Output D

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

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

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

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

$\mathsf{mIND}_b(\Pi, A, k, \ell)$:

For $i = 1, \ldots, \ell$*:* $(\mathrm{PK}_i, \mathrm{SK}_i) \leftarrow \mathsf{Gen}(1^k)$

$(\langle m_0^1, \ldots, m_0^\ell\rangle, \langle m_1^1, \ldots, m_1^\ell\rangle, \mathrm{STATE}_A) \leftarrow A_1(\langle \mathrm{PK}_1, \ldots, \mathrm{PK}_\ell\rangle)$

s.t. $|m_0^1| = |m_1^1| = \cdots = |m_0^\ell| = |m_1^\ell|$

For $i = 1, \ldots, \ell$*:* $y_i \leftarrow \mathsf{Enc}_{\mathrm{PK}_i}(m_b^i)$

$D \leftarrow A_2(y_1, \ldots, y_\ell, \mathrm{STATE}_A)$

Output D

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

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

4.2.2 Non-malleable encryption

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

$\mathsf{NME}_b(\Pi, A, k, \ell)$:

$(\mathrm{PK}, \mathrm{SK}) \leftarrow \mathsf{Gen}(1^k)$

$(m_0, m_1, \mathrm{STATE}_A) \leftarrow A_1(\mathrm{PK})$ *s.t.* $|m_0| = |m_1|$

$y \leftarrow \mathsf{Enc}_{\mathrm{PK}}(m_b)$

$(\psi_1, \ldots, \psi_\ell) \leftarrow A_2(y, \mathrm{STATE}_A)$

Output $(d_1, \ldots, d_\ell)$ *where* $d_i = \begin{cases} \perp & \text{if } \psi_i = y \\ \mathsf{Dec}_{\mathrm{SK}}(\psi_i) & \text{otherwise} \end{cases}$

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

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

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

4.2.3 (Strong) one-time signature schemes

Informally, a (strong) one-time signature scheme $(\mathsf{GenSig}, \mathsf{Sign}, \mathsf{VerSig})$ is an existentially unforgeable signature scheme, with the restriction that the signer signs at most one message with any key. This means that an efficient adversary, upon seeing a signature on a message m of his choice, cannot generate a valid signature on a different message, or a different valid signature on the same message m. Such schemes can be constructed in

a black-box way from one-way functions [R90, L79], and thus from any semantically secure encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ using black-box access only to Gen.

4.3 Construction

Given an encryption scheme $E = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$, we construct a new encryption scheme $\Pi = (\mathsf{NMGen}^{\mathsf{Gen}}, \mathsf{NMEnc}^{\mathsf{Gen},\mathsf{Enc}}, \mathsf{NMDec}^{\mathsf{Gen},\mathsf{Dec}})$, summarized in Figure 4.1, and described as follows.

Polynomial encoding. We identify $\{0,1\}^n$ with the field $\mathrm{GF}(2^n)$. To encode a message $m \in \{0,1\}^n$, we pick a random degree k polynomial p over $\mathrm{GF}(2^n)$ such that $p(0) = m$ and construct a $k \times 10k$ matrix such that the i'th column of the matrix comprise entirely of the value $s_i = p(i)$ (where $0, 1, \ldots, 10k$ are the lexicographically first $10k+1$ elements in $\mathrm{GF}(2^n)$ according to some canonical encoding). Note that $(s_1, \ldots, s_{10k})$ is both a $(k+1)$-out-of-$10k$ secret-sharing of m using Shamir's secret-sharing scheme and a codeword of the Reed-Solomon code $\mathcal{W}$, where

$$\mathcal{W} = \{\ (p(1), \ldots, p(10k) \mid p \text{ is a degree } k \text{ polynomial }\}.$$

Note that $\mathcal{W}$ is a code over the alphabet $\{0,1\}^n$ with minimum relative distance 0.9, which means we may efficiently correct up to 0.45 fraction errors using the Berlekamp-Welch algorithm.

Encryption. The public key for Π comprises $20k^2$ public keys E indexed by a triplet $(i,j,b) \in [k] \times [10k] \times \{0,1\}$; there are two keys corresponding to each entry of a $k \times 10k$ matrix. To encrypt a message m, we (a) compute $(s_1, \ldots, s_{10_k})$ as in the above-mentioned polynomial encoding, (b) generate $(\textsc{sksig}, \textsc{vksig})$ for a one-time signature, (c) compute a $k \times 10k$ matrix $\vec{c} = (c_{i,j})$ of ciphertexts where $c_{i,j} = \mathsf{Enc}_{\mathrm{PK}_{i,j}^{v_i}}(s_j)$, and (d) signs $\vec{c}$ using $\textsc{sksig}$.

$$\begin{pmatrix} \mathsf{Enc}_{\mathrm{PK}_{1,1}^{v_1}}(s_1) & \mathsf{Enc}_{\mathrm{PK}_{1,2}^{v_1}}(s_2) & \cdots & \mathsf{Enc}_{\mathrm{PK}_{1,10k}^{v_1}}(s_{10k}) \\ \mathsf{Enc}_{\mathrm{PK}_{2,1}^{v_2}}(s_1) & \mathsf{Enc}_{\mathrm{PK}_{2,2}^{v_2}}(s_2) & \cdots & \mathsf{Enc}_{\mathrm{PK}_{2,10k}^{v_2}}(s_{10k}) \\ \vdots & \vdots & \ddots & \vdots \\ \mathsf{Enc}_{\mathrm{PK}_{k,1}^{v_k}}(s_1) & \mathsf{Enc}_{\mathrm{PK}_{k,2}^{v_k}}(s_2) & \cdots & \mathsf{Enc}_{\mathrm{PK}_{k,10k}^{v_k}}(s_{10k}) \end{pmatrix}$$

Consistency Checks. A valid ciphertext in Π satisfies two properties: (1) the first row is an encryption of a codeword in $\mathcal{W}$ and (2) every column comprises k encryptions of the same plaintext. We want to design consistency checks that reject ciphertexts that are "far" from being valid ciphertexts under Π. For simplicity, we will describe the consistency checks as applied to the underlying matrix of plaintexts. The checks depend on a random subset S of k columns chosen during key generation.

Column Check (`column-check`): We check that each of the k columns in S comprises entirely of the same value.

Codeword Check (`codeword-check`): We find a codeword w that agrees with the first row of the matrix in at least $9k$ positions; the check fails if no such w exists. Then we check that the first row of the matrix agrees with w at the k positions indexed by S.

The codeword check ensures that with high probability, the first row of the matrix agrees with w in at least $10k - o(k)$ positions. We explain its significance after describing the alternative decryption algorithm in the analysis.

Decryption. To decrypt, we (a) verify the signature and run both consistency checks, and (b) if all three checks accept, decode the codeword w and output the result, otherwise output $\perp$. Note that to decrypt we only need the $20k$ secret keys corresponding to the first row of the matrix and $2k$ secret keys corresponding to each of the k columns in S.

Note that the decryption algorithm may be stream-lined, for instance, by running the codeword check only if the column check succeeds. We choose to present the algorithm as is in order to keep the analysis simple; in particular, we will run both consistency checks independent of the outcome of the other.

4.4 Analysis

Having presented our construction, we now formally state and prove our main result:

Theorem 1. (Main Theorem, restated). *Suppose there exists an encryption scheme* $E = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *that is semantically secure under a CPA attack. Then there exists an encryption scheme* $\Pi = (\mathsf{NMGen}^{\mathsf{Gen}}, \mathsf{NMEnc}^{\mathsf{Gen},\mathsf{Enc}}, \mathsf{NMDec}^{\mathsf{Gen},\mathsf{Dec}})$ *that is non-malleable under a CPA attack.*

We establish the theorem (as in [DDN00, PSV06], etc) via a series of hybrid arguments and deduce indistinguishability of the intermediate hybrid experiments from the semantic security of the underlying scheme E under some set of public keys Γ. To do so, we will need to implement an alternative decryption algorithm NMDec^* that is used in the intermediate experiments to simulate the actual decryption algorithm NMDec in the non-malleability experiment. We need NMDec^* to achieve two conflicting requirements:

- NMDec^* and NMDec must agree on essentially all inputs, including possibly malformed ciphertexts;
- We can implement NMDec^* without having to know the secret keys corresponding to the public keys in Γ.

Of course, designing NMDec^* is difficult precisely because NMDec uses the secret keys corresponding to the public keys in Γ.

Here is a high-level (but extremely inaccurate) description of how NMDec^* works: Γ is the set of public keys corresponding to the first row of the $k \times 10k$ matrix. To implement NMDec^*, we will decrypt the i'th row of the matrix of ciphertexts, for some $i > 1$, which the column check (if successful) guarantees to agree with the first row in most positions; error correction takes care of the tiny fraction of disagreements.

4.4.1 Alternative decryption algorithm NMDec^*

Let $\text{VKSIG}^* = (v_1^*, \ldots, v_k^*)$ denote the verification key in the challenge ciphertext given to the adversary in the non-malleability experiment, and let $\text{VKSIG} = (v_1, \ldots, v_k)$

denote the verification key in (one of) the ciphertext generated by the adversary. First, we modify the signature check to also output $\perp$ if there is a forgery, namely $\text{VKSIG} = \text{VKSIG}^*$. Next, we modify the consistency checks (again, as applied to the underlying matrix of plaintexts) as follows:

> COLUMN CHECK (`column-check`*): This is exactly as before, we check that the each of the k columns in S comprises entirely of the same value.
>
> CODEWORD CHECK (`codeword-check`*): Let i be the smallest value such that $v_i \neq v_i^*$ (which exists because $\text{VKSIG} \neq \text{VKSIG}^*$). We find a codeword w that agrees with the i'th row of the matrix in at least $8k$ positions (note agreement threshold is smaller than before); the check fails if so such w exists. Then we check that the first row of the matrix agrees with w at the k positions indexed by S.

To decrypt, run the modified signature and consistency checks, and if all three checks accept, decode the codeword w and output the result, otherwise output $\perp$. To implement the modified consistency checks and decryption algorithm, we only need the $10k$ secret keys indexed by $\overline{\text{VKSIG}^*}$ for each row of the matrix, and as before, the $2k$ secret keys corresponding to each of the k columns in S.

Remark on the Codeword Check. At first, the codeword check may seem superfluous. Suppose we omit the codeword check, and as before, define w to be a codeword that agrees with the first row in $9k$ positions and with the i'th row in $8k$ positions in the respective decryption algorithms; the gap is necessary to take into account inconsistencies not detected by the column check. Now, consider a malformed ciphertext ψ for Π where in the underlying matrix of plaintexts, each row is the same corrupted codeword that agrees with a valid codeword in exactly $8.5k$ positions. Without the codeword checks, ψ will be an invalid ciphertext according to NMDec and a valid ciphertext according to NMDec^* and can be used to distinguish the intermediate hybrid distributions in the analysis; with the codeword checks, ψ is an invalid ciphertext according to both. It is also easy to construct a problematic malformed ciphertext for the case where both agreement thresholds are set to the same value (say $9k$).

4.4.2 A promise problem

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

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

To quantify and establish these guarantees, we consider the following promise problem (Π_Y, Π_N) that again refers to the underlying matrix of plaintexts. An instance is a matrix of k by $10k$ values in $\{0,1\}^n \cup \perp$.

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

Π_N (NO instances) — either there exist two rows that are 0.1-far (i.e. disagree in at least k positions), or the first row is 0.1-far from every codeword in $\mathcal{W}$ (i.e. disagree with every codeword in at least k positions).

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

- If $\vec{M} \in \Pi_N$, then it fails the column/codeword checks in both decryption algorithms with high probability, in which case both decryption algorithms output $\perp$. Specifically, if there are two rows that are 0.1-far, then column check rejects $\vec{M}$ with probability $1 - 0.9^k$. On the other hand, if the first row is 0.1-far from every codeword, then the codeword check in NMDec rejects $\vec{M}$ with probability 1 and that in NMDec^* rejects

$\vec{M}$ with probability at least $1 - 0.9^k$; that is, with probability $1 - 0.9^k$, both codeword checks in NMDec and NMDec^* rejects $\vec{M}$.

– If $\vec{M} \notin \Pi_N$, then both decryption algorithms always output the same answer for all choices of the set S, provided there is no forgery. Fix $\vec{M} \notin \Pi_N$ and a set S. The first row is 0.9-close to codeword $w \in \mathcal{W}$ and we know in addition that every other row is 0.9-close to the first row and thus 0.8-close to w. Therefore, we will recover the same codeword w and message m whether we decode the first row within distance 0.1, or any other row within distance 0.2. This means that the codeword checks in both decryption algorithms compare the first row with the same codeword w. As such, both decryption algorithms output $\perp$ with exactly the same probability, and whenever they do not output $\perp$, they output the same message m.

4.4.3 Proof of main theorem

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

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

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

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

> $\mathsf{NMDec}^*_{\text{SK}}([\vec{c}, \text{VKSIG}, \sigma])$:
>
> 1. (`sig-check`*) Verify the signature with $\mathsf{VerSig}_{\text{VKSIG}}[\vec{c}, \sigma]$. Output $\perp$ if the signature fails to verify or if $\text{VKSIG} = \text{VKSIG}^*$.
> 2. Let $\vec{c} = (c_{i,j})$ and $\text{VKSIG} = (v_1, \ldots, v_k)$. Let i be the smallest value such that $v_i \neq v_i^*$. Compute $s_j = \mathsf{Dec}_{\text{SK}_{i,j}^{v_i}}(c_{i,j})$, $j = 1, \ldots, 10k$ and

$w = (w_1, \ldots, w_{10k}) \in \mathcal{W}$ that agrees with $(s_1, \ldots, s_{10k})$ in at least $8k$ positions. If no such codeword exists, output $\perp$.

3. (`column-check`*) For all $j \in S$, check that $\mathsf{Dec}_{\mathrm{SK}_{1,j}^{v_1}}(c_{1,j}) = \mathsf{Dec}_{\mathrm{SK}_{2,j}^{v_2}}(c_{2,j}) = \cdots = \mathsf{Dec}_{\mathrm{SK}_{k,j}^{v_k}}(c_{k,j})$.
4. (`codeword-check`*) For all $j \in S$, check that $\mathsf{Dec}_{\mathrm{SK}_{1,j}^{v_1}}(c_{1,j}) = w_j$.

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

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

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

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

Proof. We will show that both distributions are statistically close for all possible coin tosses in both experiments (specifically, those of NMGen, A and NMEnc) except for the choice of S in NMGen. Once we fix all the coin tosses apart from the choice of S, the output $(\psi_1, \ldots, \psi_{p(k)})$ of A_2 are completely determined and identical in both experiments. We claim that with probability $1 - 2p(k) \cdot 0.9^k = 1 - \mathrm{neg}(k)$ over the choice of S, the decryptions of $(\psi_1, \ldots, \psi_{p(k)})$ agree in both experiments. This follows from the analysis of the promise problem in Section 4.4.2. □

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

$$\left\{\mathsf{NME}_b^{(2)}(\Pi, A, k, p(k))\right\} \equiv \left\{\mathsf{mIND}_b(E, B, k, 9k^2)\right\}$$

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

– (pre-processing) Pick a random subset $S = \{u_1, \ldots, u_j\}$ of $[10k]$ and run $\mathsf{GenSig}(1^k)$ to generate $(\textsc{sksig}^*, \textsc{vksig}^*)$ and set $(v_1^*, \ldots, v_k^*) = \textsc{vksig}^*$. Let ϕ be a bijection identifying $\{(i,j) \mid i \in [k], j \in [10k] \setminus S\}$ with $[9k^2]$.

– (key generation) B receives $\langle \mathrm{PK}_1, \ldots, \mathrm{PK}_{9k^2} \rangle$ from the outside and simulates NMGen as follows: for all $i \in [k], j \in [10k], \beta \in \{0,1\}$,

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

– (message selection) Let (m_0, m_1) be the pair of messages A_1 returns. B then chooses k random values $(\gamma_{u_1}, \ldots, \gamma_{u_k}) \in \{0,1\}^n$ and computes two degree k polynomials p_0, p_1 where p_β interpolates the $k+1$ points $(0, m_\beta), (u_1, \gamma_{u_1}), \ldots, (u_k, \gamma_{u_k})$ for $\beta \in \{0,1\}$. B sets $m_\beta^{\phi(i,j)} = p_\beta(j)$, for $i \in [k], j \in [10k] \setminus S$ and forwards $(\langle m_0^1, \ldots, m_0^{9k^2} \rangle, \langle m_1^1, \ldots, m_1^{9k^2} \rangle)$ to the outside.

– (ciphertext generation) B receives $\langle y_1, \ldots, y_{9k^2} \rangle$ from the outside (according to the distribution $\mathsf{Enc}_{\mathrm{PK}_1}(m_b^1), \ldots, \mathsf{Enc}_{\mathrm{PK}_{9k^2}}(m_b^{9k^2})$) and generates a ciphertext $[\vec{c}, \mathrm{VKSIG}^*, \sigma]$ as follows:

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

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

– (decryption) Upon receiving a sequence of ciphertexts $(\psi_1, \ldots, \psi_{p(k)})$ from A_2, B decrypts these ciphertexts using NMDec^* as in $\mathsf{NME}_b^{(2)}$. Note that to simulate NMDec^*, it suffices for B to possess the secret keys $\{\mathrm{SK}_{i,j}^{\beta} \mid \beta = 1 - v_i^* \text{ or } j \in S\}$, which B generated by itself. □

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

$$\begin{aligned} \left\{\mathsf{NME}_b(\Pi, A, k, p(k))\right\} &\overset{c}{\approx} \left\{\mathsf{NME}_b^{(1)}(\Pi, A, k, p(k))\right\} \\ &\overset{s}{\approx} \left\{\mathsf{NME}_b^{(2)}(\Pi, A, k, p(k))\right\} \equiv \left\{\mathsf{mIND}_b(E, B, k, 9k^2)\right\} \end{aligned}$$

By Prop 1, $\mathsf{mIND}_0(E, B, k, 9k^2) \overset{c}{\approx} \mathsf{mIND}_1(E, B, k, 9k^2)$, which concludes the proof of Theorem 1.

4.5 Achieving Bounded-CCA2 Non-Malleability

We sketch how our scheme may be modified to achieve non-malleability under a bounded-CCA2 attack. Here, we allow the adversary to query Dec at most q times in the non-malleability experiment (but it must not query Dec on y). The modification is the straight-forward analogue of the [CHH+07] modification of the [PSV06] scheme: we increase the number of columns in the matrix from $10k$ to $80(k+q)$, and the degree of the polynomial p and the size of S from k to $8(k+q)$, and propagate the changes accordingly. The analysis is basically as before, except for the following claim (where $\mathsf{NME\text{-}q\text{-}CCA}_b^{(1)}, \mathsf{NME\text{-}q\text{-}CCA}_b^{(2)}$ are the respective analogues of $\mathsf{NME}_b^{(1)}, \mathsf{NME}_b^{(1)}$):

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

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

Proof (sketch). As before, we will show that both distributions are statistically close for all possible coin tosses in both experiments (specifically, those of NMGen, A and NMEnc) except for the choice of S in NMGen. However, we cannot immediately deduce that the output of A_2 are completely determined and identical in both experiments, since they depend on the adaptively chosen queries to NMDec, and the answers depend on S. Instead, we will consider all 2^q possible computation paths of A which are determined based on the q query/answer pairs from NMDec. For each query, we consider the underlying matrix of plaintexts $\vec{M}$:

- If $\vec{M} \in \Pi_N$, then we assume NMDec returns $\perp$.
- If $\vec{M} \notin \Pi_N$, then we consider two branches depending on the two possible outcomes of the consistency checks.

We claim that with probability $1 - 2^q \cdot p(k) \cdot 0.9^{8(k+q)} > 1 - \mathrm{neg}(k)$ over the choice of S, the decryptions of $(\psi_1, \ldots, \psi_{p(k)})$ agree in both experiments in all 2^q computation paths. $\square$

Remark on achieving (full) CCA2 security. It should be clear from the preceding analysis that the barrier to obtaining full CCA2 security lies in handling queries outside Π_N. Specifically, with even just a (full) CCA1 attack, an adversary could query NMDec on a series of adaptively chosen ciphertexts corresponding to matrices outside Π_N to learn the set S upon which it could readily break the security of our construction.

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

1. For $i \in [k], j \in [10k], b \in \{0,1\}$, run $\mathsf{Gen}(1^k)$ to generate key-pairs $(\text{PK}^b_{i,j}, \text{SK}^b_{i,j})$.
2. Pick a random subset $S \subset [10k]$ of size k.

Set $\text{PK} = \left\{(\text{PK}^0_{i,j}, \text{PK}^1_{i,j}) \mid i \in [k], j \in [10k]\right\}$, $\text{SK} = \left\{S, (\text{SK}^0_{i,j}, \text{SK}^1_{i,j}) \mid i \in [k], j \in [10k]\right\}$.

$\mathsf{NMEnc}_{\text{PK}}(m)$:

1. Pick random $\alpha_1, \ldots, \alpha_k \in \text{GF}(2^n)$ and set $s_j = p(j), j \in [10k]$ where $p(x) = m_0 + \alpha_1 x + \ldots + \alpha_k x^k$.
2. Run $\mathsf{GenSig}(1^k)$ to generate $(\text{SKSIG}, \text{VKSIG})$. Let $(v_1, \ldots, v_k)$ be the binary representation of VKSIG.
3. Compute the ciphertext $c_{i,j} \leftarrow \mathsf{Enc}_{\text{PK}^{v_i}_{i,j}}(s_j)$, for $i \in [k], j \in [10k]$.
4. Compute the signature $\sigma \leftarrow \mathsf{Sign}_{\text{SKSIG}}(\vec{c})$ where $\vec{c} = (c_{i,j})$.

Output the tuple $[\vec{c}, \text{VKSIG}, \sigma]$.

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

1. (`sig-check`) Verify the signature with $\mathsf{VerSig}_{\text{VKSIG}}[\vec{c}, \sigma]$.
2. Let $\vec{c} = (c_{i,j})$ and $\text{VKSIG} = (v_1, \ldots, v_k)$. Compute $s_j = \mathsf{Dec}_{\text{SK}^{v_1}_{1,j}}(c_{1,j})$, $j = 1, \ldots, 10k$ and the codeword $w = (w_1, \ldots, w_{10k}) \in \mathcal{W}$ that agrees with $(s_1, \ldots, s_{10k})$ in at least $9k$ positions. If no such codeword exists, output $\perp$.
3. (`column-check`) For all $j \in S$, check that $\mathsf{Dec}_{\text{SK}^{v_1}_{1,j}}(c_{1,j}) = \mathsf{Dec}_{\text{SK}^{v_2}_{2,j}}(c_{2,j}) = \cdots = \mathsf{Dec}_{\text{SK}^{v_k}_{k,j}}(c_{k,j})$.
4. (`codeword-check`) For all $j \in S$, check that $s_j = w_j$.

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

Figure 4.1: The Non-Malleable Encryption Scheme Π

Chapter 5

Future Directions

Many interesting questions on the black-box complexity of cryptographic primitives remain to be answered.

Lower bounds on efficiency. Many cryptographic constructions based on one-way functions are significantly more efficient under the additional assumption that the function is a permutation. This pertains not just to statistically hiding commitments, but also to the construction of pseudorandom generators [BM84, Y82, HILL99] as well as hardness amplification [Y82, GIL+90]. In each of these cases, the constructions have roughly linear complexity from permutations (as measured by round, query, and randomness complexity) and polynomial complexity from arbitrary functions.

The long-standing failure (since the early 1990s) to improve the constructions from arbitrary functions to match the efficiency of those from permutations suggests that there is some fundamental barrier. In the case of pseudorandom generators, a result of Reingold et al. [RTV04] shows that if such improved constructions exist, then they can be also realized with a weakly black-box construction. As such, even though weakly black-box constructions do capture some of the limitations of one-way permutations (c.f. [GT00]), they are too broad to account for the gaps. There are two consequences to this result: if we are looking for positive results, then we may treat the one-way function as an oracle in the construction, but we must somehow exploit the code of the adversary in the proof of security, except we do not know any non-black-box technique that works in this way. The other is that if

we want to show negative results to account for the failure of current techniques, then we should turn to more restrictive classes of constructions, such as fully black-box ones.

There is indeed reason to believe a super-linear lower bound for these problems for fully black-box construction: we have toy constructions with linear complexity which are certainly insecure as fully black-box constructions but the security of which are uncertain as weakly black-box constructions. For instance, consider a derandomized direct product construction for hardness amplification, e.g., using a pairwise independent generator based on affine functions over large fields. It is easy to see that this construction cannot be proven secure with a black-box proof of security, but we do not know if it is secure with a non-black-box proof of security. We hope that the techniques introduced in Chapter 3 for exploiting black-box proofs of security, along with the follow-up work of Haitner et al. [HHRS07] will help establish super-linear lower bounds for fully black-box constructions, at least in some special cases of these problems.

Extensions to the black-box model. A common non-black-box usage of a cryptographic primitive involves applying an NP-reduction to a statement that refers to the primitive (e.g. [GMW87, DDN00, PSV06]). An interesting research direction is to look into relaxations of a black-box construction that admit such constructions. The very recent notion of algebrization introduced by Aaronson and Wigderson [AW07] might be relevant here.

Bibliography

[AH91] W. Aiello and J. Håstad. Statistical zero-knowledge languages can be recognized in two rounds. *JCSS*, 42(3):327–345, 1991.

[AIK06] B. Applebaum, Y. Ishai, and E. Kushilevitz. Cryptography in NC^0. *SIAM J. Comput.*, 36(4):845–888, 2006.

[AW07] S. Aaronson and A. Wigderson. Algebrization: A new barrier in complexity theory. Manuscript, 2007. `http://www.scottaaronson.com/papers/`.

[B01] B. Barak. How to go beyond the black-box simulation barrier. In *FOCS*, pages 106–115, 2001.

[BCY91] G. Brassard, C. Crépeau, and M. Yung. Constant-round perfect zero-knowledge computationally convincing protocols. *Theoretical Computer Science*, 84(1):23–52, 1991.

[BFM88] M. Blum, P. Feldman, and S. Micali. Non-interactive zero-knowledge and its applications. In *STOC*, pages 103–112, 1988.

[BGW88] M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *STOC*, pages 1–10, 1988.

[BHZ87] R. B. Boppana, J. Håstad, and S. Zachos. Does co-NP have short interactive proofs? *IPL*, 25(2):127–132, 1987.

[BJY97] M. Bellare, M. Jakobsson, and M. Yung. Round-optimal zero-knowledge arguments based on any one-way function. In *EUROCRYPT*, pages 280–305, 1997.

[BM84] M. Blum and S. Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SICOMP*, 13(4):850–864, 1984.

[CCM98] C. Cachin, C. Crépeau, and J. Marcil. Oblivious transfer with a memory-bounded receiver. In *FOCS*, pages 493–502, 1998.

[CDMW08] S. G. Choi, D. Dachman-Soled, T. Malkin, and H. Wee. Black-box construction of a non-malleable encryption scheme from any semantically secure one. In *TCC*, 2008.

[CHH+07] R. Cramer, G. Hanaoka, D. Hofheinz, H. Imai, E. Kiltz, R. Pass, A. Shelat, and V. Vaikuntanathan. Bounded CCA2-secure encryption. In *ASIACRYPT*, 2007.

[CHK04] R. Canetti, S. Halevi, and J. Katz. Chosen-ciphertext security from identity-based encryption. In *EUROCRYPT*, pages 207–222, 2004.

[CS98] R. Cramer and V. Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In *CRYPTO*, pages 13–25, 1998.

[CS02] R. Cramer and V. Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In *EUROCRYPT*, pages 45–64, 2002.

[CS06] C. Crépeau and G. Savvides. Optimal reductions between oblivious transfers using interactive hashing. In *EUROCRYPT*, pages 201–221, 2006.

[D93] I. Damgård. Interactive hashing can simplify zero-knowledge protocol design without computational assumptions. In *CRYPTO*, pages 100–109, 1993.

[DDN00] D. Dolev, C. Dwork, and M. Naor. Nonmalleable cryptography. *SIAM J. Comput.*, 30(2):391–437, 2000.

[DGOW95] I. Damgård, O. Goldreich, T. Okamoto, and A. Wigderson. Honest verifier vs dishonest verifier in public coin zero-knowledge proofs. In *CRYPTO*, pages 325–338, 1995.

[DGW95] I. Damgård, O. Goldreich, and A. Wigderson. Information theory versus complexity theory: Another test case. manuscript, 1995.

[DHRS04] Y. Z. Ding, D. Harnik, A. Rosen, and R. Shaltiel. Constant-round oblivious transfer in the bounded storage model. In *TCC*, pages 446–472, 2004.

[DPP98] I. Damgård, T. P. Pedersen, and B. Pfitzmann. Statistical secrecy and multibit commitments. *IEEE Transactions on Information Theory*, 4(3):1143–1151, 1998.

[ES02] E. Elkind and A. Sahai. A unified methodology for constructing public-key encryption schemes secure against adaptive chosen-ciphertext attack. Cryptology ePrint Archive, Report 2002/024, 2002. `http://eprint.iacr.org/`.

[F89] L. Fortnow. The complexity of perfect zero-knowledge. *Advances in Computing Research*, 5:429–442, 1989.

[F02] M. Fischlin. On the impossibility of constructing non-interactive statistically-secret protocols from any trapdoor one-way function. In *CT-RSA*, pages 79–95, 2002.

[FS89] U. Feige and A. Shamir. Zero knowledge proofs of knowledge in two rounds. In *CRYPTO*, pages 526–544, 1989.

[GGKT05] R. Gennaro, Y. Gertner, J. Katz, and L. Trevisan. Bounds on the efficiency of generic cryptographic constructions. *SIAM Journal on Computing*, 35(1):217–246, 2005.

[GIL+90] O. Goldreich, R. Impagliazzo, L. A. Levin, R. Venkatesan, and D. Zuckerman. Security preserving amplification of hardness. In *FOCS*, pages 318–326, 1990.

[GK96a] O. Goldreich and A. Kahan. How to construct constant-round zero-knowledge proof systems for NP. *J. Cryptology*, 9(3):167–190, 1996.

[GK96b] O. Goldreich and H. Krawczyk. On the composition of zero-knowledge proof systems. *SIAM Journal on Computing*, 25(1):169–192, 1996.

[GM84] S. Goldwasser and S. Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28(2):270–299, 1984.

[GMM07] Y. Gertner, T. Malkin, and S. Myers. Towards a separation of semantic and CCA security for public key encryption. In *TCC*, pages 434–455, 2007.

[GMR89] S. Goldwasser, S. Micali, and C. Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, 18(1):186–208, 1989.

[GMR01] Y. Gertner, T. Malkin, and O. Reingold. On the impossibility of basing trapdoor functions on trapdoor predicates. In *FOCS*, pages 126–135, 2001.

[GMW87] O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game or a completeness theorem for protocols with honest majority. In *STOC*, pages 218–229, 1987.

[GSV98] O. Goldreich, A. Sahai, and S. P. Vadhan. Honest-verifier statistical zero-knowledge equals general statistical zero-knowledge. In *STOC*, pages 399–408, 1998.

[GT00] R. Gennaro and L. Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *FOCS*, pages 305–313, 2000.

[H08] I. Haitner. Semi-honest to malicious oblivious transfer - the black-box way. In *TCC*, 2008.

[HHK+05] I. Haitner, O. Horvitz, J. Katz, C.-Y. Koo, R. Morselli, and R. Shaltiel. Reducing complexity assumptions for statistically-hiding commitment. In *EUROCRYPT*, pages 58–77, 2005.

[HHRS07] I. Haitner, J. Hoch, O. Reingold, and G. Segev. Finding collisions in interactive protocols – a tight lower bound on the round complexity of statistically-hiding commitments. In *FOCS*, pages 669–679, 2007.

[HHS08] I. Haitner, J. Hoch, and G. Segev. A linear lower bound on the communication complexity of single-server private information retrieval. In *TCC*, 2008.

[HILL99] J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

[HK05] O. Horvitz and J. Katz. Bounds on the efficiency of "black-box" commitment schemes. In *Proc. 32nd ICALP*, 2005.

[HN06] D. Harnik and M. Naor. On the compressibility of NP instances and cryptographic applications. In *FOCS*, pages 719–728, 2006.

[HR07a] I. Haitner and O. Reingold. A new interactive hashing theorem. In *IEEE Conference on Computational Complexity*, pages 319–332, 2007.

[HR07b] I. Haitner and O. Reingold. Statistically-hiding commitment from any one-way function. In *STOC*, pages 1–10, 2007.

[IKLP06] Y. Ishai, E. Kushilevitz, Y. Lindell, and E. Petrank. Black-box constructions for secure computation. In *STOC*, pages 99–108, 2006.

[IL89] R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography (extended abstract). In *FOCS*, pages 230–235, 1989.

[IR89] R. Impagliazzo and S. Rudich. Limits on the provable consequences of one-way permutations. In *STOC*, pages 44–61, 1989.

[KS06] T. Koshiba and Y. Seri. Round-efficient one-way permutation based perfectly concealing bit commitment scheme. ECCC TR06-093, 2006.

[KST99] J. H. Kim, D. R. Simon, and P. Tetali. Limits on the efficiency of one-way permutation-based hash functions. In *FOCS*, pages 535–542, 1999.

[L79] L. Lamport. Constructing digital signatures from a one-way function. Technical Report SRI-CSL-98, SRI International Computer Science Laboratory, 1979.

[L06] Y. Lindell. A simpler construction of CCA2-secure public-key encryption under general assumptions. *J. Cryptology*, 19(3):359–377, 2006.

[LP07] Y. Lindell and B. Pinkas. An efficient protocol for secure two-party computation in the presence of malicious adversaries. In *EUROCRYPT*, pages 52–78, 2007.

[LTW05] H. Lin, L. Trevisan, and H. Wee. On hardness amplification of one-way functions. In *TCC*, pages 34–49, 2005.

[NOV06] M.-H. Nguyen, S. J. Ong, and S. P. Vadhan. Statistical zero-knowledge arguments for NP from any one-way function. In *FOCS*, pages 3–14, 2006.

[NOVY98] M. Naor, R. Ostrovsky, R. Venkatesan, and M. Yung. Perfect zero-knowledge arguments for NP using any one-way permutation. *J. Cryptology*, 11(2):87–108, 1998.

[NV06] M.-H. Nguyen and S. P. Vadhan. Zero knowledge with efficient provers. In *STOC*, pages 287–295, 2006.

[NY89] M. Naor and M. Yung. Universal one-way hash functions and their cryptographic applications. In *STOC*, pages 33–43, 1989.

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

[OVY93] R. Ostrovsky, R. Venkatesan, and M. Yung. Fair games against an all-powerful adversary. In *AMS DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 1993.

[OW93] R. Ostrovsky and A. Wigderson. One-way fuctions are essential for non-trivial zero-knowledge. In *ISTCS*, 1993.

[PSV06] R. Pass, A. Shelat, and V. Vaikuntanathan. Construction of a non-malleable encryption scheme from any semantically secure one. In *CRYPTO*, pages 271–289, 2006.

[PW07] C. Peikert and B. Waters. Lossy trapdoor functions and their applications. Cryptology ePrint Archive, Report 2007/279, 2007. `http://eprint.iacr.org/`.

[R90] J. Rompel. One-way functions are necessary and sufficient for secure signatures. In *STOC*, pages 387–394, 1990.

[R91] S. Rudich. The use of interaction in public cryptosystems. In *CRYPTO*, pages 242–251, 1991.

[RS91] C. Rackoff and D. R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In *CRYPTO*, pages 433–444, 1991.

[RTV04] O. Reingold, L. Trevisan, and S. Vadhan. Notions of reducibility between cryptographic primitives. In *TCC*, pages 1–20, 2004.

[S98] D. R. Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? In *EUROCRYPT*, pages 334–345, 1998.

[S99] A. Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *FOCS*, pages 543–553, 1999.

[W07] H. Wee. One-way permutations, interactive hashing and statistically hiding commitments. In *TCC*, pages 419–433, 2007.

[Y82] A. C.-C. Yao. Theory and applications of trapdoor functions (extended abstract). In *FOCS*, pages 80–91, 1982.

[Y86] A. C.-C. Yao. How to generate and exchange secrets (extended abstract). In *FOCS*, pages 162–167, 1986.