48th Annual IEEE Symposium on Foundations of Computer Science

Finding Collisions in Interactive Protocols – A Tight Lower Bound on the Round Complexity of Statistically-Hiding Commitments*

Iftach Haitner Jonathan J. Hoch Omer Reingold† Gil Segev

Department of Computer Science and Applied Mathematics
Weizmann Institute of Science, Israel

Abstract

We study the round complexity of various cryptographic protocols. Our main result is a tight lower bound on the round complexity of any fully-black-box construction of a statistically-hiding commitment scheme from one-way permutations, and even from trapdoor permutations. This lower bound matches the round complexity of the statistically-hiding commitment scheme due to Naor, Ostrovsky, Venkatesan and Yung (CRYPTO '92). As a corollary, we derive similar tight lower bounds for several other cryptographic protocols, such as single-server private information retrieval*,* interactive hashing*, and* oblivious transfer *that guarantees statistical security for one of the parties.*

Our techniques extend the collision-finding oracle due to Simon (EUROCRYPT '98) to the setting of interactive protocols (our extension also implies an alternative proof for the main property of the original oracle). In addition, we substantially extend the reconstruction paradigm of Gennaro and Trevisan (FOCS '00). In both cases, our extensions are quite delicate and may be found useful in proving additional black-box separation results.

1 Introduction

Research in the foundations of cryptography is concerned with the construction of provably secure cryptographic tools. The security of such constructions relies on a growing number of computational assumptions, and in the last few decades much research has been devoted to demonstrating the feasibility of particular cryptographic tasks based on the weakest possible assumptions. For example, the existence of one-way functions has been shown to be equivalent to the existence of pseudorandom functions and permutations [22, 45], pseudorandom generators [3, 33], universal one-way hash functions and signature schemes [49, 54], different types of commitment schemes [32, 33, 46, 50], private-key encryption [21] and other primitives.

Many constructions based on minimal assumptions, however, result in only a theoretical impact due to their inefficiency, and in practice more efficient constructions based on seemingly stronger assumptions are being used. Thus, identifying tradeoffs between the *efficiency* of cryptographic constructions and the strength of the computational assumptions on which they rely is essential in order to obtain a better understanding of the relationship between cryptographic tasks and computational assumptions.

In this paper we follow this line of research, and study the tradeoffs between the *round complexity* of cryptographic protocols and the strength of their underlying computational assumptions. We provide a lower bound on the round complexity of black-box constructions of statistically-hiding and computationally-binding commitment schemes (for short, statistical commitment schemes) based on one-way permutations and on families of trapdoor permutations. Our lower bound matches known upper bounds resulting from [47]. As a corollary of our main result, we derive similar tight lower bounds for several other cryptographic protocols, such as *single-server private information retrieval*, *interactive hashing*, and *oblivious transfer* that guarantees statistical security for one of the parties. In the following paragraphs, we discuss the notion of statistically-hiding commitment schemes and describe the setting in which our lower bounds are proved.

Statistically-hiding commitment schemes. A commitment scheme defines a two-stage interactive protocol between a sender $\mathcal{S}$ and a receiver $\mathcal{R}$; informally, after the *commit stage*, $\mathcal{S}$ is bound to (at most) one value, which stays hidden from $\mathcal{R}$, and in the *reveal stage* $\mathcal{R}$ learns this value. The two security properties hinted at in this informal description are known as *binding* ($\mathcal{S}$ is bound to at most one value after the commit stage) and *hiding* ($\mathcal{R}$ does not learn the value to which $\mathcal{S}$ commits before the reveal stage). In

*A longer version, including proofs of all claims, appears as [29].

†Research supported by grant 1300/05 from the Israel Science Foundation.

0272-5428/07 $25.00 © 2007 IEEE
DOI 10.1109/FOCS.2007.7

a statistical commitment scheme the hiding property holds *even against all-powerful receivers* (i.e., the hiding holds information-theoretically), while the binding property is required to hold only for polynomially-bounded senders.

Statistical commitments can be used as a building block in constructions of statistical zero-knowledge arguments [5, 47] and of certain coin-tossing protocols [42]. When used within protocols in which certain commitments are never revealed, statistical commitments have the following advantage over computationally-hiding commitment schemes: in such a scenario, it should be infeasible to violate the binding property *only during the execution of the protocol*, whereas the committed values will remain hidden *forever* (i.e., regardless of how much time the receiver invests after the completion of the protocol).

Statistical commitments schemes with a constant number of rounds were shown to exist based on specific number-theoretic assumptions [4, 5] (or, more generally, based on any collection of claw-free permutations [26] with an efficiently-recognizable index set [23]), and collision-resistant hash functions [10, 49]. Protocols with higher round complexity were shown to exist based on different types of one way functions. Protocols with $O\left(\frac{n}{\log n}\right)$ rounds (where n is the input length of the underlying function) were based on one-way permutations [47] and (known-) regular one-way functions [30].[1] Finally, a protocol with a polynomial number of rounds was based on any one-way function [32, 50].

Black-box reductions. As already mentioned, we are interested in proving lower bounds on the round complexity of various cryptographic constructions. In particular, we are interested in showing that any construction of statistical commitments based on trapdoor permutations requires a fairly large number of rounds. Nevertheless, under standard assumptions such as the existence of collision-resistant hash functions, *constant-round statistical commitments do exist*. So if these assumptions hold, then the existence of trapdoor permutations implies the existence of constant-round statistical commitments in a *trivial logical sense*. Faced with similar difficulties, Impagliazzo and Rudich [35] presented a paradigm for proving impossibility results under a restricted, yet important, subclass of reductions called *black-box reductions*. Intuitively, a black-box reduction of a primitive P to a primitive Q is a construction of P out of Q that ignores the internal structure of the implementation of Q and just uses it as a "subroutine" (i.e., as a black-box). In addition, in the case of fully-black-box reductions, the proof of security (showing that an adversary that breaks the implementation of P implies an adversary that breaks the implementation of Q), is also black-box (i.e., the internal structure of the adversary that breaks the implementation of P is ignored as well). For a more exact treatment of black-box reductions we refer the reader to the full version of this paper.

[1] The original presentations of the above protocols have $O(n)$ rounds. By a natural extension, however, the number of rounds in these protocols can be reduced to $O\left(\frac{n}{\log n}\right)$, see [31, 39].

1.1 Related Work

Impagliazzo and Rudich [35] showed that there are no black-box reductions of key-agrement protocols to one-way permutations and substantial additional work in this line followed (c.f. [18, 56, 57]). Kim, Simon and Tetali [38] initiated a new line of impossibility results, by providing a lower bound on the *efficiency* of black-box reductions (rather than on their feasibility). They proved a lower bound on the efficiency, in terms of the number of calls to the underlying primitive, of any black-box reduction of universal one-way hash functions to one-way permutations. This result was later improved, to match the known upper bound, by Gennaro et al. [16], which also provided tight lower bounds on the efficiency of several other black-box reductions [14, 15, 16]. Building upon the technique developed by [16], Horvitz and Katz [34] gave lower bounds on the efficiency of black-box reductions of statistically-binding commitments to one-way permutations. In all the above results the measure of efficiency under consideration is the number of calls to the underlying primitives.

With respect to the round complexity of statistical commitments, Fischlin [13] showed that every black-box reduction of statistical commitments to trapdoor permutations, has at least two rounds. His result follows Simon's oracle separation of collision-resistant hash functions from one-way permutations [57]. Recently, Wee [58] considered a restricted class of black-box reductions of statistical commitments to one-way permutations. Informally, Wee considered only constructions in which the sender first queries the one-way permutation on several independent inputs. Once the interaction with the receiver starts, the sender only access the outputs of these queries (and not the inputs) and does not perform any additional queries. Wee showed that every black-box reduction of the above class has $\Omega\left(\frac{n}{\log n}\right)$ communication rounds.

The study of lower bounds on the round complexity of black-box reductions, was also addressed in the context of zero-knowledge protocols [7, 11, 24, 27, 37, 55]. In this context, however, the black-box access is to the, possibly cheating, verifier and not to an underlying primitive.

1.2 Our Results

We study the class of fully-black-box constructions of statistically-hiding commitment schemes from trapdoor permutations, and prove a lower bound on the round complexity of any such construction. Informally, our main theorem is as follows:

Main Theorem (Informal). *Any fully-black-box construction of a statistically-hiding commitment scheme from a family of trapdoor permutations over* $\{0,1\}^n$ *has* $\Omega\left(\frac{n}{\log n}\right)$ *communication rounds.*

In fact, we consider a more general notion of hardness for trapdoor permutations, which extends the standard polynomial hardness requirement. Informally, we say that a trapdoor permutation τ over $\{0,1\}^n$ is $s(n)$-hard if any probabilistic Turing-machine that runs in time $s(n)$ inverts τ on a uniformly chosen image with probability at most $1/s(n)$. Given this definition, we show that any fully-black-box construction of a statistically-hiding commitment scheme from a family of $s(n)$-hard trapdoor permutations over $\{0,1\}^n$ requires $\Omega\left(\frac{n}{\log s(n)}\right)$ communication rounds.

Our lower bound, for both notions of trapdoor permutations, matches the known upper bound due to [47, 31, 39]. The scheme of Naor et al. relies on one-way permutations in a fully-black-box manner, and thus we demonstrate that their scheme is essentially optimal with respect to the number of communication rounds. Moreover, our lower bound implies that trapdoor permutations are not superior to one-way permutations in this setting, whereas collision-resistant hash functions and specific number-theoretic assumptions are superior and imply constant-round schemes.

Taking the security of the reduction into account. Note that the informal statement of our main theorem considers constructions which invoke only trapdoor permutations over n bits. We would like to extend the result to consider constructions which may invoke the trapdoor permutations over more than a single domain. However, in this case, better upper bounds are known. In particular, given security parameter 1^n it is possible to apply the scheme of Naor et al. using a one-way permutation over n^ϵ bits. This implies statistical commitments that run in $O(n^\epsilon)$ rounds. This subtle issue is not unique to our setting, and in fact arises in any study of the efficiency of cryptographic reductions (see, in particular, [16, 58]). The common approach for addressing this issue is by restricting the class of constructions (as in the informal statement of our main theorem above). In Section 4 we follow a less restrictive approach: we consider constructions which are given access to trapdoor permutations over *any* domain size, but require that the proof of security will be "somewhat security preserving". More specifically, we consider an additional parameter, which we refer to as the *security-parameter-expansion* of the construction. Informally, the proof of security in a fully-black-box construction gives a way to translate (in a black-box manner) an adversary S^* that breaks the binding of the commitment scheme into an adversary A that breaks the security of the trapdoor permutation. Such a construction is $\ell(n)$-security-parameter-expanding if whenever the machine A tries to invert a permutation over n bits, it invokes S^* on security parameters which are at most $1^{\ell(n)}$. It should be noted that any construction in which $\ell(n)$ is significantly larger than n, may only be weakly security preserving (for a taxonomy of security preserving reductions see [44, Lecture 2]).

Our lower bound proof takes into consideration the security parameter expansion, and therefore our statements apply for the most general form of fully-black-box reductions. In particular, in case that $\ell(n) = O(n)$, our theorem implies that the required number of rounds is $\Omega\left(\frac{n}{\log n}\right)$, and in the general case (where $\ell(n)$ may be any polynomial in n), our theorem implies that the required number of rounds is $n^{\Omega(1)}$ (which as argued above is tight as well).

Implications to other cryptographic protocols. Our main result can be extended to any cryptographic protocol which implies statistically-hiding commitment schemes in a fully-black-box manner, as long as the reduction essentially preserves the number of communication rounds. Specifically, we derive similar $\Omega\left(\frac{n}{\log n}\right)$ lower bounds on the round complexity of fully-black-box constructions from trapdoor permutations of single-server private information retrieval, interactive hashing, and oblivious transfer that guarantees statistical security for one of the parties.

1.3 Overview of the Technique

For simplicity, we focus in this overview on the round complexity lower bound for statistical commitment which are based on one-way permutations (the lower bound for constructions based on families of trapdoor permutations follows similar ideas). We also assume without loss of generality that the sender's secret in the commitment protocol is a single uniform bit. Let us start by considering Simon's oracle [57] for ruling out any black-box reduction of collision resistant hash functions to one-way permutation.

Simon's oracle. Simon's oracle $\mathsf{ColFinder}$ gets as an input a circuit C, possibly with π gates,[2] where π is a random permutation. It then outputs two random elements w_1 and w_2 such that $C(w_1) = C(w_2)$. Clearly, in the presence of $\mathsf{ColFinder}$ no family of collision resistant hash functions exists (the adversary simply queries $\mathsf{ColFinder}$ with the hash function circuit to find a collision). In order to rule out the existence, in the presence of $\mathsf{ColFinder}$, of any two-round statistical commitment scheme, Fischlin [13] used the following adversary $\mathcal{S}^*$ to break any such scheme: assume w.l.o.g. that the first message, q_1 is sent by $\mathcal{R}$ and consider the circuit C_{q_1}, naturally defined by q_1 and $\mathcal{S}$. Namely, C_{q_1} gets as an input the random coins of $\mathcal{S}$ and outputs the answer that $\mathcal{S}$ replies on receiving the message q_1 from $\mathcal{R}$. In the commit stage after receiving the message q_1, the cheating $\mathcal{S}^*$ constructs C_{q_1}, queries $\mathsf{ColFinder}(C_{q_1})$ to get w_1 and w_2, and answers as $\mathcal{S}(w_1)$ would (i.e., by $C_{q_1}(w_1)$).

[2] In fact, $\mathsf{ColFinder}$ also accepts circuits C with $\mathsf{ColFinder}$ gates. For the sake of this discussion, we ignore this property.

In the reveal stage, $\mathcal{S}^*$ uses both w_1 and w_2 to open the commitment (i.e. once using the random coins w_1 and then using w_2). Since the protocol is statistically hiding, the set of the sender's random coins that are consistent with this commit stage transcript is divided to almost equal size parts by the values of their secret bits. Therefore, with probability roughly half w_1 and w_2 will differ on the value of $\mathcal{S}$'s secret bit and the binding of the commitment will be violated.

In order to obtain the black-box impossibility results (both of [57] and of [13]), it is left to show that π is one-way in the presence of ColFinder. Let A be a circuit trying to invert π on a random $y \in \{0,1\}^n$ using ColFinder, and lets assume for now that A makes only a single call to ColFinder. Intuitively, the way we could hope this query to ColFinder with input C could help is by "hitting" y in the following sense: we say that ColFinder *hits* y on input C, if the computations of $C(w_1)$ or of $C(w_2)$ query π on $\pi^{-1}(y)$. Now we note that for every input circuit C each one of w_1 and w_2 (the outputs of ColFinder on C) is *individually* uniform. Therefore, the probability that ColFinder hits y on input C, may only be larger by a factor two than the probability that evaluating C on a uniform w queries π on $\pi^{-1}(y)$. In other words, A does not gain much by querying ColFinder (as A can evaluate C on a uniform w on its own). Formalizing the above intuition is far from easy, mainly when we consider A that queries ColFinder more than once. The difficulty lies in formalizing the claim that the only useful queries are the ones in which ColFinder hits y (after all, the reply to a query may give us some useful global information on π). We give some intuition in Section 1.4 for why this claim is valid, following a different approach than the original proof due to [57].

Finding collisions in interactive protocols. We would like to employ Simon's oracle for breaking the binding of more interactive protocols (with more than two rounds). Unfortunately, the "natural" attempts to do so seem to fail miserably. The first attempt that comes to mind might be the following: In the commit stage $\mathcal{S}^*$ follows the protocol and let $q_1, \ldots, q_k$ be the messages that $\mathcal{R}$ sent in this stage. In the reveal stage, $\mathcal{S}^*$ queries ColFinder to get a colliding pair (w_1, w_2) in $C_{q_1,\ldots,q_k}$ - the circuit naturally defined by the code of $\mathcal{S}$ and $q_1, \ldots, q_k$ (i.e., $C_{q_1,\ldots,q_k}$ gets as an input the random coins of $\mathcal{S}$ and outputs the messages sent by $\mathcal{S}$ when $\mathcal{R}$'s messages are $q_1, \ldots, q_k$). The problem is that it is very unlikely that the outputs of Sam on $C_{q_1,\ldots,q_k}$ will be consistent with the answers that $\mathcal{S}^*$ *already* gave in the commit stage (we did not encounter this problem when breaking two-round protocols, since $\mathcal{S}^*$ could query ColFinder on C_{q_1} before $\mathcal{S}^*$ sends its first and only message). Alternatively, we could change ColFinder such that it gets as an additional input w_1 and returns w_2 for which $C_{q_1,\ldots,q_k}(w_1) = C_{q_1,\ldots,q_k}(w_2)$ (that is, the new ColFinder finds second preimages rather than collisions). Indeed, this new ColFinder does imply the breaking of any commitment scheme, but it also implies the inversion of π.[3] We should not be too surprised that both the above attempts failed as they are both completely oblivious of the round complexity of $(\mathcal{S}, \mathcal{R})$. Since one-way permutations *do imply* statistical commitments in a black-box manner any oracle that breaks statistical commitments could also be used to break the underlying one-way permutations.[4]

For our oracle separation, we manage to extend Simon's oracle to the setting of interactive protocols. We will have to handle interaction with care so that our oracle is not too strong (so that it does not break the one-way permutations), but still strong enough to be useful. In fact, the more interactive our oracle will be the more powerful it will be, and eventually it will allow breaking the one-way permutations. Quantifying this growth in power is how we get our tight bounds on the round complexity of the reduction.

Our oracle. It will be useful for us to view Simon's oracle as performing two sampling tasks: First it samples w_1 uniformly and then it samples a second preimage w_2 such that $C(w_1) = C(w_2)$. As explained above, an oracle for sampling a second preimage allows inverting the one-way permutations. The reason the sampling done by ColFinder is not too damaging is that w_1 *was chosen by* ColFinder *after* C *is already given.* Therefore, an adversary A is very limited in setting up the second distribution from which ColFinder samples (i.e. the uniform distribution over the preimages of $C(w_1)$ under C). In other words, this distribution is jointly defined by A and ColFinder itself.

Extending the above interpretation of ColFinder (and ignoring various technical aspects), our separation oracle Sam is defined as follows: Sam will be given as input a query $Q = (C_{\text{next}}, C, z)$, and will output a pair (w', z') where w' is a uniformly distributed preimage of z under the mapping defined by the circuit C, and $z' = C_{\text{next}}(w')$. Following the intuition above we impose the restriction that there was a previous query $(C, \cdot, \cdot)$ that was answered by (w, z) (note that this imposes a forest-like structure on the queries). In other words, C was announced *before* w *was chosen by* Sam *in answering the previous query.*[5] In addition, we only allow querying Sam up to depth $d(n)+1$ where n is the security parameter (this depth function $d(\cdot)$ will depend on the

[3] Consider a circuit C, whose input is composed of a bit σ and an n-bit string w. The circuit C is defined by $C(0, w) = \pi(w)$ and $C(1, w) = w$. Thus, in order to compute $\pi^{-1}(y)$ we can simply invoke the new ColFinder on input C and $w_1 = (1, y)$. With probability half ColFinder will return $w_2 = (0, \pi^{-1}(y))$.

[4] In addition, in both these naive attempts the cheating sender $\mathcal{S}^*$ follows the commit stage honestly (as $\mathcal{S}$ would). It is not hard to come up with two-round protocol that works well for semi-honest commit stage senders (consider for instance the two-round variant of [47] where the receiver's queries are all sent in the first round).

[5] An additional important restriction that we will not discuss here is that C_{next} is a *refinement* of the circuit C, where by refinement we mean that $C_{\text{next}}(w) = (C(w), \widetilde{C}(w))$ for some circuit $\widetilde{C}$ and for every w.

particular lower bound we will try to prove).

Sam allows breaking $d(n)$-round statistical commitments. The adversary $\mathcal{S}^*$ operates as follows: after getting the first message q_1, it constructs C_{q_1} (the circuit that computes $\mathcal{S}$'s first message) and queries Sam for a random input w_1 (i.e., it queries Sam without specifying C and z), and sends $\mathcal{R}$ the message specified by $z_1 = C_{q_1}(w_1)$. On getting the i-th receiver message q_i, the adversary $\mathcal{S}^*$ constructs $C_{q_1,\ldots,q_i}$ (the circuit that computes $\mathcal{S}$'s first i messages), queries Sam on $(C_{q_1,\ldots,q_i}, C_{q_1,\ldots,q_{i-1}}, z_{i-1})$ to get (w_i, z_i), and replies to $\mathcal{R}$ with the message specified by $z_i = C_{q_1,\ldots,q_i}(w_i)$. Finally, after completing the commit stage (when answering the last receiver message q_d) it queries Sam on $(\perp, C_{q_1,\ldots,q_d}, z_d)$ to get w_{d+1}, z_{d+1}. Both w_d and w_{d+1} are random inputs that are consistent with the commit-stage transcript. Thus, with probability roughly half they can be used to break the binding of the protocol.

Sam cannot be used to invert random permutations. To complete our impossibility result, it is left to prove that Sam cannot be used to invert the random permutation π. As in our intuition for Simon's oracle, we would like to claim that the only useful Sam-queries for an adversary A that tries to invert π on y are queries that make Sam hit y. Assume Sam is given as input a query (C_{next}, C, z), and outputs a pair (w', z'). We say that Sam hits y if evaluating $C(w')$ queries π on $\pi^{-1}(y)$. Extending the reconstruction technique of Gennaro and Trevisan, we show that A is unlikely to invert π on y if it does not make Sam hit y (see Section 1.4).

The most technical part of the paper is showing that a circuit A that inverts π on y while making Sam hit y can be transformed into a circuit M that inverts π without Sam hitting y. This aspect of the proof is somewhat influenced by the work of Wee [58]. Let us try to give some intuition for this claim. Assume for simplicity of notation that A only makes the following queries: $(C_1, \perp, \perp)$, $(C_2, C_1, z_1), \ldots, (C_{d+1}, C_d, z_d)$ and it receives the corresponding replies: $(w_1, z_1), \ldots, (w_{d+1}, z_{d+1})$. We know that for some i the probability that the computation $C_i(w_{i+1})$ queries π on $\pi^{-1}(y)$ (i.e., hits y) is non-negligible (as we know that Sam is likely to hit y). On the other hand the probability that $C_1(w_2)$ hits y (which is identical to the probability that $C_1(w_1)$ hits y) is exponentially small. Therefore, unless $d = \Omega\left(\frac{n}{\log n}\right)$ we have that there exists a location i such that the probability $C_i(w_{i+1})$ hits y is larger than the probability that $C_{i-1}(w_i)$ hits y by a very large polynomial. We are also able to show (under the various restrictions on Sam) that the probability that the computation $C_i(w_i)$ hits y is unlikely to be much smaller than the probability that the computation $C_i(w_{i+1})$ hits y. Combining the above understandings we design M that inverts π on y with non-negligible probability without making Sam hit y (and this will constitute a contradiction). M simulates A but in addition, whenever A queries Sam for (C_{i+1}, C_i, z_i) and receives a reply (w_{i+1}, z_{i+1}) we let M also evaluate $C_{i+1}(w_{i+1})$. If this computation queries π on $\pi^{-1}(y)$ then M halts and outputs $\pi^{-1}(y)$. Otherwise, M continues with the simulation of A. We argue that with sufficiently large probability, if the first query of A that makes Sam hit y is (C_{i+1}, C_i, z_i), then M's computation of $C_i(w_i)$ queries π on $\pi^{-1}(y)$. Therefore, M retrieves $\pi^{-1}(y)$ before making the hitting query.

1.4 Extending Gennaro and Trevisan's reconstruction lemma

Gennaro and Trevisan [16] presented a very elegant argument for proving that a random permutation is hard to invert for non-uniform adversaries (previous proofs, e.g. [35], only ruled out uniform adversaries). Let A be a circuit and let π be a permutation that A inverts on a non-negligible fraction of its outputs. Gennaro and Trevisan showed that relative to A the permutation π has a relatively short description. Therefore, by a counting argument, there is only a tiny fraction of permutations which A inverts well. Intuitively, A saves on the description of π as it allows us to reconstruct π on (many of) the x's for which $A^\pi(\pi(x)) = x$. The formal proof strongly relies on a bound on the number of π gates in A: when we use A to reconstruct π on x we need all the π-queries made by $A^\pi(\pi(x))$ (apart perhaps of the query for $\pi(x)$ itself) to already be reconstructed.

In our setting, we would like to consider an adversary $A^{\mathsf{Sam}}(y)$ that (many times) inverts y without making Sam produce a y-hit. Recall that the oracle Sam is given as an input a circuit C with π-gates and has to produce a random inverse of some value z under the mapping defined by C. We would like to apply the argument of [16] to claim that relative to A and Sam there is a short description of π. However, we are faced with a substantial obstacle as the simulation of Sam requires making a huge amount of π queries.[6] Overcoming this obstacle requires much care both in the definition and analysis of Sam, and we refer the reader to the full version of this paper for more details.

Paper organization. In Section 2, we briefly present the notations and formal definitions used in this paper. In Section 3 we describe the oracle that is used to derive our results. In Section 4 we state the main properties of our oracle, which are then combined to derive our main result. Finally, Section 5 discusses the implications of the result.

2 Preliminaries

We denote by Π_n the set of all permutations over $\{0,1\}^n$. The statistical distance between two distributions X and Y over Ω is denoted $\mathrm{SD}(X, Y)$.

[6] Consider for example C such that on input w it truncates the last bit of $\pi(w)$ and outputs the result. Finding collisions in C requires knowledge of π almost entirely.

One-way permutations and trapdoor permutations. We briefly present the notions of one-way permutations and trapdoor permutations which are used in this paper. For a more comprehensive discussion we refer the reader to [19].

Definition 2.1. A collection of permutations $\pi = \{\pi_n\}_{n=1}^{\infty}$, where $\pi_n \in \Pi_n$ for every n, is $s(n)$-*hard* if for every probabilistic Turing-machine A that runs in time $s(n)$, and for all sufficiently large n,

$$\Pr\left[A(1^n, y) = \pi_n^{-1}(y)\right] \leq \frac{1}{s(n)} \ ,$$

where the probability is taken uniformly over all the possible choices of $y \in \{0,1\}^n$ and over all the possible outcomes of the internal coin tosses of A.

A collection of trapdoor permutations is represented as a triplet $\tau = \left(G, F, F^{-1}\right)$. Informally, G corresponds to a key generation procedure, which is queried on a string td (intended as the "trapdoor") and produces a corresponding public key pk. The procedure F is the actual permutation, which is queried on a public key pk and an input x. Finally, the procedure F^{-1} is the inverse of F: If $G(td) = pk$ and $F(pk, x) = y$, then $F^{-1}(td, y) = x$. In this paper, since we are concerned with providing a lower bound, we do not consider the most general definition of a collection of trapdoor permutations. Instead, we denote by T_n the set of all triplets $\tau_n = \left(G_n, F_n, F_n^{-1}\right)$ of the following form:

1. $G_n \in \Pi_n$.
2. $F_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ is a function such that $F_n(pk, \cdot) \in \Pi_n$ for every $pk \in \{0,1\}^n$.
3. $F_n^{-1} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ is a function such that $F_n^{-1}(td, y)$ returns the unique $x \in \{0,1\}^n$ for which $F_n(G_n(td), x) = y$.

Our lower bound proof is based on analyzing random instances of such collections. A uniformly distributed $\tau_n \in T_n$ can be chosen as follows: G_n is chosen uniformly at random from Π_n, and for each $pk \in \{0,1\}^n$ a permutation $F_n(pk, \cdot)$ is chosen uniformly and independently at random from Π_n.

Definition 2.2. A family of trapdoor permutations $\tau = \left\{\tau_n = \left(G_n, F_n, F_n^{-1}\right)\right\}_{n=1}^{\infty}$ is $s(n)$-*hard* if for every probabilistic Turing-machine A that runs in time $s(n)$, and for all sufficiently large n,

$$\Pr\left[A^{\tau}(1^n, G_n(td), y) = F_n^{-1}(td, y)\right] \leq \frac{1}{s(n)} \ ,$$

where the probability is taken uniformly over all the possible choices of $td \in \{0,1\}^n$ and $y \in \{0,1\}^n$, and over all the possible outcomes of the internal coin tosses of A.

Note that Definition 2.2 refers to the difficulty of inverting a random permutation $F(pk, \cdot)$ on a uniformly distributed image y, when given only $pk = G(td)$ and y. Some applications, however, require enhanced hardness conditions. For example, it may be required (cf. [20, Appendix C]) that it is hard to invert $F(pk, \cdot)$ on y even given the random coins used in the generation of y. Note that our formulation captures such hardness condition as well and therefore the impossibility results proved in this paper hold also for enhanced trapdoor permutations.[7]

Commitment schemes. In this paper, where we are interested in proving an impossibility result for commitment schemes, it will be sufficient for us to deal with bit-commitment schemes. A bit-commitment scheme is defined via a triplet of probabilistic polynomial-time Turing-machines $(\mathcal{S}, \mathcal{R}, \mathcal{V})$ such that:

- $\mathcal{S}$ receives as input the security parameter 1^n and a bit b. Following its interaction, it outputs some information dec (the decommitment).
- $\mathcal{R}$ receives as input the security parameter 1^n. Following its interaction, it outputs a state information com (the commitment).
- $\mathcal{V}$ (acting as the receiver in the reveal stage[8]) receives as input the security parameter 1^n, a commitment com and a decommitment dec, and outputs a bit b' or $\perp$.

Denote by $(\mathsf{dec}|\mathsf{com}) \leftarrow \langle \mathcal{S}(1^n, b), \mathcal{R}(1^n) \rangle$ the experiment in which $\mathcal{S}$ and $\mathcal{R}$ interact (using the given inputs and uniformly chosen random coins), and then $\mathcal{S}$ outputs dec while $\mathcal{R}$ outputs com. It is required that for all n, every bit b, and every pair $(\mathsf{dec}|\mathsf{com})$ that may be output by $\langle \mathcal{S}(1^n, b), \mathcal{R}(1^n) \rangle$, it holds that $\mathcal{V}(\mathsf{com}, \mathsf{dec}) = b$.[9]

In order to define the security properties of *statistically-hiding* commitment schemes, we first introduce the following notation. Given a commitment scheme $(\mathcal{S}, \mathcal{R}, \mathcal{V})$ and a Turing-machine $\mathcal{R}^*$, we denote by $\mathsf{view}_{\langle \mathcal{S}(b), \mathcal{R}^* \rangle}(n)$ the distribution on the view of $\mathcal{R}^*$ when interacting with $\mathcal{S}(1^n, b)$. This view consists of $\mathcal{R}^*$'s random coins and of the sequence of messages it receives from $\mathcal{S}$. The distribution is taken over the random coins of both $\mathcal{S}$ and $\mathcal{R}$. We note that the following definition describes a relatively weak hiding requirement, in which the sender is protected only against the honest receiver. However, since we prove a lower bound, this only strengthens our result.

[7] A different enhancement, used by [28], requires the permutations' domain to be polynomially dense in $\{0,1\}^n$. Clearly, our impossibility result holds w.r.t. this enhancement as well.

[8] Note that there is no loss of generality in assuming that the reveal stage is non-interactive. This is since any such interactive stage can be replaced with a non-interactive one as follows: The sender sends its internal state to the receiver, who then simulates the sender in the interactive stage.

[9] Although we assume perfect completeness, it is not essential for our results.

Definition 2.3. A bit-commitment scheme $(\mathcal{S}, \mathcal{R}, \mathcal{V})$ is *honest-receiver* $\rho(n)$*-hiding* if the ensembles $\{\mathsf{view}_{\langle \mathcal{S}(0), \mathcal{R}\rangle}(n)\}$ and $\{\mathsf{view}_{\langle \mathcal{S}(1), \mathcal{R}\rangle}(n)\}$ have statistical difference at most $\rho(n)$ for all sufficiently large n. Such a scheme is *honest-receiver statistically-hiding* if it is honest-receiver $\rho(n)$-hiding for a negligible function $\rho(n)$.

Definition 2.4. A bit-commitment scheme $(\mathcal{S}, \mathcal{R}, \mathcal{V})$ is $\mu(n)$*-binding* if for every probabilistic polynomial-time Turing-machine $\mathcal{S}^*$ it holds that

$$\Pr\left[\begin{array}{c}((\mathsf{dec}, \mathsf{dec}')|\mathsf{com}) \leftarrow \langle \mathcal{S}^*(1^n), \mathcal{R}(1^n)\rangle : \\ \mathcal{V}(\mathsf{com}, \mathsf{dec}) = 0, \mathcal{V}(\mathsf{com}, \mathsf{dec}') = 1\end{array}\right] < \mu(n)$$

for all sufficiently large n, where the probability is taken over the random coins of both $\mathcal{S}^*$ and $\mathcal{R}$. Such a scheme is *computationally-binding* if it is $\mu(n)$-binding for some negligible function $\mu(n)$, and is *weakly-binding* if it is $(1 - 1/p(n))$-binding for some polynomial $p(n)$.

Black-box reductions. A reduction of a primitive P to a primitive Q is a construction of P out of Q. Such a reduction is *semi-black-box* if it ignores the internal structure of Q's implementation, and it is *fully-black-box* if the proof of correctness is black-box as well. A taxonomy of black-box reductions was provided by Reingold, Trevisan and Vadhan [53], and the reader is referred to their paper for a more complete and formal view of these notions. We focus here on a specific definition for our setting.

Definition 2.5. A fully-black-box construction of a weakly-binding honest-receiver statistically-hiding commitment scheme from an $s(n)$-hard family of trapdoor permutations is a quadruple of probabilistic oracle Turing-machines $(\mathcal{S}, \mathcal{R}, \mathcal{V}, A)$ for which the following hold:

1. **Correctness:** For every family τ of trapdoor permutations, $(\mathcal{S}^\tau, \mathcal{R}^\tau, \mathcal{V}^\tau)$ is an honest-receiver statistically-hiding commitment scheme.
2. **Black-box proof of binding:** For every family $\tau = \left\{\tau_n = \left(G_n, F_n, F_n^{-1}\right)\right\}_{n=1}^{\infty}$ of trapdoor permutations and for every probabilistic polynomial-time Turing-machine $\mathcal{S}^*$, if $\mathcal{S}^*$ with oracle access to τ breaks the weak binding of $(\mathcal{S}^\tau, \mathcal{R}^\tau, \mathcal{V}^\tau)$, then

$$\Pr\left[A^{\tau, \mathcal{S}^*}(1^n, G_n(td), y) = F_n^{-1}(td, y)\right] > \frac{1}{s(n)} \ ,$$

for infinitely many values of n, where A runs in time $s(n)$, and the probability is taken uniformly over all the possible choices of $td \in \{0,1\}^n$ and $y \in \{0,1\}^n$, and over all the possible outcomes of the internal coin tosses of A.

We remark that the above correctness requirement is very strict and is not essential for our results. In fact, for every τ such that the protocol $(\mathcal{S}^\tau, \mathcal{R}^\tau, \mathcal{V}^\tau)$ is a statistically-hiding commitment scheme, we construct a malicious sender $\mathcal{S}^*$ which breaks the binding property of the scheme. Therefore, we could have dealt with a weaker correctness requirement as well, but stating such a weaker requirement in a meaningful way turns out to be quite subtle.

In addition, it would be useful for us to consider the following property of fully-black-box constructions: Consider a malicious sender $\mathcal{S}^*$ that breaks the binding of the commitment scheme and consider the machine A that wishes to break the security of the trapdoor permutation. Then, A receives a security parameter 1^n and invokes $\mathcal{S}^*$ in a black-box manner. Definition 2.5, however, does not restrict the range of security parameters that A is allowed to invoke $\mathcal{S}^*$ on. For example, A may invoke $\mathcal{S}^*$ on security parameter 1^{n^2}, or even on security parameter $1^{\Theta(s(n))}$, where $s(n)$ is the running time of A. The following definition will enable us to capture this property of the construction, and again, we present a specific definition for our setting.

Definition 2.6. A fully-black-box construction $(\mathcal{S}, \mathcal{R}, \mathcal{V}, A)$ is $\ell(n)$*-security-parameter-expanding*, if for every malicious sender $\mathcal{S}^*$, the machine A on security parameter 1^n invokes $\mathcal{S}^*$ on security parameters which are at most $1^{\ell(n)}$.

3 The Oracle

In this section we describe the oracle that implies our lower bound. Our oracle $\mathcal{O}$ is of the form $(\tau, \mathsf{Sam}^\tau)$, where τ is a family of trapdoor permutations (i.e., $\tau = \{\tau_n\}_{n=1}^{\infty}$, where τ_n is a trapdoor permutation over $\{0,1\}^n$), and Sam^τ is an oracle that, very informally, receives as input a description of a circuit C (which may contain τ-gates) and a string z, and outputs a uniformly distributed preimage of z under the mapping defined by C. As discussed in the introduction, we will impose several essential restrictions on the querying of Sam that will prevent it from assisting in inverting τ.

Description of Sam. The oracle Sam receives as input a query $Q = (C^\tau_{\text{next}}, C^\tau, z)$, and outputs a pair (w', z') where w' is a uniformly distributed preimage of z under the mapping defined by the circuit C^τ, and $z' = C^\tau_{\text{next}}(w')$. We impose the following restrictions:

1. z was the result of a previous query with C^τ as the next-query circuit (note that this imposes a forest-like structure on the queries).
2. The circuit C^τ_{next} is a *refinement* of the circuit C^τ, where by a refinement we mean that $C^\tau_{\text{next}}(w) = (C^\tau(w), \widetilde{C}^\tau(w))$ for some circuit $\widetilde{C}^\tau$ and for every w. In particular, this implies that C^τ and C^τ_{next} have the same input length. Given a query Q, we denote this input length by $m(Q)$, and when the query Q is clear from the context we will write only m.

3. Each query contains a security parameter 1^n, and Sam answers queries only up to depth $\mathsf{depth}(n)$, for some "depth restriction" function $\mathsf{depth} : \mathbb{N} \to \mathbb{N}$ which is part of the description of Sam. The security parameter is set such that a query with security parameter 1^n is allowed to contain circuits with queries to permutations on up to n bits. Note that although different queries may have different security parameters, we ask that in the same "query-tree", all queries will have the same security parameter (hence the depth of the tree is already determined by the root query).

In order to impose these restrictions, we equip Sam with a family $\mathsf{sign} = \{\mathsf{sign}_k\}_{k=1}^{\infty}$ of (random) functions $\mathsf{sign}_k : \{0,1\}^k \to \{0,1\}^{2k}$ that will be used as "signatures" for identifying legal queries as follows: in addition to outputting (w', z'), Sam will also output the value $\mathsf{sign}(1^n, C^\tau_{\mathrm{next}}, z', dep+1)$, where dep is the depth of the query, 1^n is the security parameter of the query, and by applying the "function" sign we actually mean that we apply the function sign_k for the correct input length. Each query of the form $Q = (1^n, C^\tau_{\mathrm{next}}, C^\tau, z, dep, sig)$ is answered by Sam if and only if C^τ_{next} is a refinement of C^τ, $dep \leq \mathsf{depth}(n)$ and $sig = \mathsf{sign}(1^n, C^\tau, z, dep)$.

Finally, we provide Sam with a family of (random) permutations $\mathcal{F} = \{f_Q\}$, where for every possible query Q a permutation f_Q is chosen uniformly at random from $\Pi_{m(Q)}$. Given a query $Q = (1^n, C^\tau_{\mathrm{next}}, C^\tau, z, dep, sig)$, the oracle Sam uses the permutation $f_Q \in \mathcal{F}$ in order to sample w' as follows: it outputs $w' = f_Q(t)$ for the lexicographically smallest $t \in \{0,1\}^m$ such that $C^\tau(f_Q(t)) = z$. Note that whenever the permutation f_Q is chosen from Π_m uniformly at random, and independently of all other permutations in $\mathcal{F}$, then w' is indeed a uniformly distributed preimage of z. In this paper, whenever we consider the probability of an event over the choice of the family $\mathcal{F}$, we mean that for each query Q a permutation f_Q is chosen uniformly at random from $\Pi_{m(Q)}$ and independently of all other permutations. A formal description of the oracle is provided in Figure 1.

On input $Q = (1^n, C^\tau_{\mathrm{next}}, C^\tau, z, dep, sig)$, the oracle $\mathsf{Sam}^{\tau,\mathcal{F},\mathsf{sign}}_{\mathsf{depth}}$ acts as follows:

1. If $C^\tau = \perp$, then output (w', z', sig') where $w' = f_Q(0^m)$, $z' = C^\tau_{\mathrm{next}}(w')$, and $sig' = \mathsf{sign}(1^n, C^\tau_{\mathrm{next}}, z', 1)$.
2. Else, if C^τ_{next} is a refinement of C^τ, $dep \leq \mathsf{depth}(n)$ and $sig = \mathsf{sign}(1^n, C^\tau, z, dep)$, then
   (a) Find the lexicographically smallest $t \in \{0,1\}^m$ such that $C^\tau(f_Q(t)) = z$.
   (b) Output (w', z', sig') where $w' = f_Q(t)$, $z' = C^\tau_{\mathrm{next}}(w')$, and $sig' = \mathsf{sign}(1^n, C^\tau_{\mathrm{next}}, z', dep+1)$.
3. Else, output $\perp$.

Figure 1: The oracle Sam.

As mentioned above, the restrictions impose a forest-like structure on any sequence of queries: each query of the form $Q = (1^n, C^\tau_{\mathrm{next}}, \perp, \perp, \perp, \perp)$ serves as a root of a tree. For any other "legal" query $Q = (1^n, C^\tau_{\mathrm{next}}, C^\tau, z, dep, sig)$, there exists a previous query Q' which resulted in output z and contained C^τ as its next-query circuit. The query Q' is identified as the parent of Q in the query forest and is denoted $Q' = \mathrm{p}(Q)$. If there is more than one such Q', then we choose the first Q' according to some fixed ordering of the queries. When dealing with Turing-machines, we can identify the queries according to their chronological order.[10]

Notation 3.1. We say that A queries $\mathsf{Sam}^{\tau,\mathcal{F},\mathsf{sign}}_{\mathsf{depth}}$ up to depth d, if for every query $Q = (1^n, C^\pi_{\mathrm{next}}, C^\pi, z, dep, sig)$ that A makes to Sam, it holds that $dep \leq d$.

4 The Round Complexity Lower Bound

In this section we first state the main properties of Sam, and then combine them to present a formal statement of the lower bound. Due to space limitation, the reader is referred to the full version of this paper for the complete proofs. The first property is that a random instance of Sam can be used to break the binding of any statistically-hiding commitment scheme. More specifically, for every statistically-hiding commitment scheme $(\mathcal{S}, \mathcal{R}, \mathcal{V})$ with oracle access to a family of trapdoor permutations, we construct a malicious sender $\mathcal{S}^*$ which has oracle access to $\mathsf{Sam}^{\tau,\mathcal{F},\mathsf{sign}}_{\mathsf{depth}}$, and breaks the binding of $(\mathcal{S}^\tau, \mathcal{R}^\tau, \mathcal{V}^\tau)$ with high probability over the choices of τ, $\mathcal{F}$ and sign. The key point is that if the commitment scheme has $d(n)$ communication rounds, then $\mathcal{S}^*$ needs to query Sam only up to depth $d(n)+1$.

Theorem 4.1. *For every $d(n)$-round honest-receiver statistically-hiding bit-commitment scheme $(\mathcal{S}, \mathcal{R}, \mathcal{V})$ with oracle access to a family of trapdoor permutations, there exist a polynomial-time malicious sender $\mathcal{S}^*$ and a negligible function $\nu(n)$, such that*

$$\Pr\left[\begin{array}{c}((\mathsf{dec}, \mathsf{dec}')|\mathsf{com}) \leftarrow \left\langle \mathcal{S}^{*\,\mathsf{Sam}^{\tau,\mathcal{F},\mathsf{sign}}_{d+1}}(1^n), \mathcal{R}^\tau(1^n)\right\rangle : \\ \mathcal{V}^\tau(\mathsf{com}, \mathsf{dec}) = 0, \mathcal{V}^\tau(\mathsf{com}, \mathsf{dec}') = 1\end{array}\right] > 1 - \nu(n)\ ,$$

for all sufficiently large n, where the probability is taken uniformly over τ, $\mathcal{F}$, sign and over $\mathcal{R}$'s random coins.

The second property is that any circuit with oracle access to Sam that tries to invert a random trapdoor permutation, fails with high probability. More specifically, we relate this success probability to the maximal depth of the Sam-queries made by the circuit, and to the size of the circuit.

[10] However, when dealing with circuits we will have to identify the queries according to a some topological order.

Theorem 4.2. *For every circuit A of size at most $s(n)$ that queries* Sam *up to depth $d(n)$ such that $s(n)^{3d(n)+2} < 2^{n/8}$, for every depth restriction function* depth *and for all sufficiently large n, it holds that*

$$\Pr\left[A^{\tau,\mathsf{Sam}^{\tau,\mathcal{F},\mathsf{sign}}_{\mathsf{depth}}}(G_n(td), y) = F_n^{-1}(td, y)\right] \leq \frac{2}{s(n)} \ ,$$

where the probability is taken uniformly over $td \leftarrow \{0,1\}^n$, $y \leftarrow \{0,1\}^n$, τ, $\mathcal{F}$ and sign.

As described in Section 1.2, given a fully-black-box construction of a statistically-hiding commitment scheme from trapdoor permutations, we consider three parameters:

1. $d(n)$ – the number of communication rounds in the commitment scheme with security parameter 1^n.
2. $s(n)$ – the hardness of the trapdoor permutation family (see Definition 2.2).
3. $\ell(n)$ – the security parameter expansion of the construction (see Definition 2.6).

We first state our result for the more standard hardness notion of trapdoor permutations, in which we consider a family of trapdoor permutations which is $s(n)$-hard for any polynomial $s(n)$. As discussed in Section 1.2, we consider both constructions which are security-preserving (i.e., $\ell(n) = O(n)$), and constructions which are not necessarily security-preserving (i.e., $\ell(n)$ is any polynomial in n). We begin by formally stating our results for these two cases.

Theorem 4.3. *Any $O(n)$-security-parameter-expanding fully-black-box construction of a weakly-binding and honest-receiver statistically-hiding commitment scheme from a family of trapdoor permutations has $\Omega\left(\frac{n}{\log n}\right)$ communication rounds.*

Theorem 4.4. *Any fully-black-box construction of a weakly-binding and honest-receiver statistically-hiding commitment scheme from a family of trapdoor permutations has $n^{\Omega(1)}$ communication rounds.*

The above two theorems are in fact obtained as corollaries of a more general statement. In this statement we specifically consider the notion of $s(n)$-hard trapdoor permutations, and do not consider a particular range for the security parameter expansion $\ell(n)$. The following is the formal statement of our main theorem, which is proved in the full version:

Theorem 4.5 (Main Theorem). *For every $\ell(n)$-security-parameter-expanding fully-black-box construction of a $d(n)$-round weakly-binding and statistically-hiding commitment scheme from an $s(n)$-hard family of trapdoor permutations, it holds that $d(\ell(n)) = \Omega\left(\frac{n}{\log s(n)}\right)$.*

5 Implications to Other Cryptographic Protocols

Our lower bound on the round complexity of statistical commitment schemes implies similar lower bounds for several other cryptographic protocols. Our result can be extended to any cryptographic protocol which can be used to construct a weakly-binding honest-receiver statistically-hiding commitment scheme in a fully-black-box manner. Specifically, in this section we derive new lower bounds on the round complexity of single-server private information retrieval, interactive hashing, and oblivious transfer that guarantees statistical security for one of the parties. We state the corollaries in this section for construction that are security preserving (i.e., $O(n)$-security-parameter-expanding) and note that more general statements, as in Theorem 4.5, could be easily derived as well.

Single-Server Private Information Retrieval. A single-server private information retrieval (PIR) scheme [9] is a protocol between a server and a user. The server holds a database $x \in \{0,1\}^n$, and the user holds an index $i \in [n]$ to an entry of the database. Informally, the user wishes to retrieve the i-th entry of the database, without revealing to the server the value i. A naive solution is to have the user download the entire database, however, the total communication complexity of this solution is n bits. Based on specific number-theoretic assumptions, several schemes with sublinear communication complexity were developed (see [6, 8, 17, 43, 40], and a recent survey by Ostrovsky and Skeith [51]). The only non-trivial construction based on general computational assumptions is due to Kushilevitz and Ostrovsky [41]. Assuming the existence of trapdoor permutations, they constructed an interactive protocol whose communication complexity is $n - o(n)$ bits.

Beimel, Ishai, Kushilevitz and Malkin [2] showed that any single-server PIR protocol with communication complexity of at most $n/2$ bits, can be used to construct a weakly-binding statistically-hiding commitment scheme. Their construction is both fully-black-box and preserves the number of rounds. Thus, by combining this with our result, we obtain the following corollary:

Corollary 5.1. *Any $O(n)$-security-parameter-expanding fully-black-box construction of a single-server PIR protocol for an n-bit database from a family of trapdoor permutations, in which the server communicates less than $n/2$ bits, has $\Omega\left(\frac{n}{\log n}\right)$ communication rounds.*

Corollary 5.1 yields in particular a lower bound on the communication complexity: any such construction requires the server to communicate $\Omega\left(\frac{n}{\log n}\right)$ bits.

Interactive Hashing. Interactive hashing was introduced by Naor, Ostrovsky, Venkatesan and Yung [47] and is a protocol that allows a sender $\mathcal{S}$ to commit to a value y while

only revealing to the receiver $\mathcal{R}$ the value $(h, z = h(y))$, where h is a 2-to-1 hash function chosen interactively during the protocol.[11] The two security properties of interactive hashing are binding ($\mathcal{S}$ is bounded by the protocol to producing at most one value of y which is consistent with the transcript) and hiding ($\mathcal{R}$ does not obtain any information about y, except for $h(y)$). Naor et al. constructed an interactive hashing protocol from any one-way permutation with $O\left(\frac{n}{\log n}\right)$ communication rounds, and showed that it implies in a fully-black-box manner a statistical commitment scheme with the same number of rounds.[12] Wee [58] has recently showed that a restricted class of fully-black-box constructions of interactive hashing from one-way permutations has $\Omega\left(\frac{n}{\log n}\right)$ rounds. Our result extends Wee's lower bound both to include the most general form of such constructions, and to trapdoor permutations.

Corollary 5.2. *Any $O(n)$-security-parameter-expanding fully-black-box construction of an interactive hashing protocol from a family of trapdoor permutations has $\Omega\left(\frac{n}{\log n}\right)$ communication rounds.*

Oblivious Transfer. Oblivious transfer (OT), introduced by Rabin [52], is a fundamental primitive in cryptography. In particular, it was shown to imply secure multiparty computation [25, 36, 60]. OT has several equivalent formulations, and we consider the formulation of $\binom{2}{1}$-OT, defined by Even, Goldreich and Lempel [12]. $\binom{2}{1}$-OT is a protocol between two parties, a sender and a receiver. The sender's input consists of two secret bits (b_0, b_1), and the receiver's input consists of a value $i \in \{0, 1\}$. At the end of the protocol, the receiver should learn the bit b_i while the sender does not learn the value i. The security of the protocol guarantees that even a cheating receiver should not be able to learn the bit b_{1-i}, and a cheating sender should not be able to learn i.

Given any $\binom{2}{1}$-OT protocol that guarantees statistical security for one of the parties (sender or receiver), one can construct a weakly-binding statistically-hiding commitment scheme in a fully-black-box manner while preserving the number of rounds. For the explicit reduction from a statistically protected sender see [13], and the reduction from a statistically protected receiver follows similar lines.[13] Thus, by combining this with our result, we obtain the following corollary:

Corollary 5.3. *Any $O(n)$-security-parameter-expanding fully-black-box construction of a $\binom{2}{1}$-OT protocol that guarantees statistical security for one of the parties from a family of trapdoor permutations has $\Omega\left(\frac{n}{\log n}\right)$ communication rounds.*

We stress that there exist constructions of semi-honest receiver $\binom{2}{1}$-OT protocols, relying on specific number-theoretic assumptions, where the sender enjoys statistical security with a constant number of rounds (e.g., Aiello et al. [1] and Naor and Pinkas [48]). Hence, as for statistical commitments, we demonstrate a large gap between the round complexity of OT constructions based on general assumptions and OT constructions based on specific number-theoretic assumptions.

[11] Several extensions to this definition were suggested, see [31, 50].

[12] Although the original proof in [47] showed the result for $O(n)$ rounds, this was recently reduced to $O\left(\frac{n}{\log n}\right)$ rounds [31, 39].

[13] Alternatively, refer to [59] for switching the roles of the sender and the receiver.

References

[1] W. Aiello, Y. Ishai, and O. Reingold. Priced oblivious transfer: How to sell digital goods. *EUROCRYPT '01*, pp. 119–135, 2001.

[2] A. Beimel, Y. Ishai, E. Kushilevitz, and T. Malkin. One-way functions are essential for single-server private information retrieval. *31st STOC*, pp. 89–98, 1999.

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

[4] J. Boyar, S. A. Kurtz, and M. W. Krentel. A discrete logarithm implementation of perfect zero-knowledge blobs. *J. of Crypto.*, 2(2):63–76, 1990.

[5] G. Brassard, D. Chaum, and C. Crépeau. Minimum disclosure proofs of knowledge. *JCSS*, 37(2):156–189, 1988.

[6] C. Cachin, S. Micali, and M. Stadler. Computationally private information retrieval with polylogarithmic communication. *EUROCRYPT '99*, pp. 402–414, 1999.

[7] R. Canetti, J. Kilian, E. Petrank, and A. Rosen. Black-box concurrent zero-knowledge requires (almost) logarithmically many rounds. *SICOMP*, 32(1):1–47, 2002.

[8] Y. Chang. Single database private information retrieval with logarithmic communication. *9th ACISP*, pp. 50–61, 2004.

[9] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan. Private information retrieval. *36th FOCS*, pp. 41–50, 1995.

[10] I. Damgård, T. P. Pedersen, and B. Pfitzmann. On the existence of statistically hiding bit commitment schemes and fail-stop signatures. *J. of Crypto.*, 10(3):163–194, 1997.

[11] C. Dwork, M. Naor, and A. Sahai. Concurrent zero-knowledge. *JACM*, 51(6):851–898, 2004.

[12] S. Even, O. Goldreich, and A. Lempel. A randomized protocol for signing contracts. *CACM*, 28(6):637–647, 1985.

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

[14] R. Gennaro, Y. Gertner, and J. Katz. Lower bounds on the efficiency of encryption and digital signature schemes. *35th STOC*, pp. 417–425, 2003.

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

[16] R. Gennaro and L. Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. *41st FOCS*, pp. 305–313, 2000.

[17] C. Gentry and Z. Ramzan. Single-database private information retrieval with constant communication rate. *32nd ICALP*, pp. 803–815, 2005.

[18] Y. Gertner, S. Kannan, T. Malkin, O. Reingold, and M. Viswanathan. The relationship between public key encryption and oblivious transfer. *41st FOCS*, pp. 325–335, 2000.

[19] O. Goldreich. *Foundations of Cryptography – Volume 1: Basic Tools*. Cambridge University Press, 2001.

[20] O. Goldreich. *Foundations of Cryptography – Volume 2: Basic Applications*. Cambridge University Press, 2004.

[21] O. Goldreich, S. Goldwasser, and S. Micali. On the cryptographic applications of random functions. *CRYPTO '84*, pp. 276–288, 1984.

[22] O. Goldreich, S. Goldwasser, and S. Micali. How to construct random functions. *JACM*, 33(4):792–807, 1986.

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

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

[25] O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game or a completeness theorem for protocols with honest majority. *19th STOC*, pp. 218–229, 1987.

[26] S. Goldwasser, S. Micali, and R. L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SICOMP*, 17(2):281–308, 1988.

[27] S. Hada and T. Tanaka. On the existence of 3-round zero-knowledge protocols. *CRYPTO '98*, pp. 408–423, 1998.

[28] I. Haitner. Implementing oblivious transfer using collection of dense trapdoor permutations. *1st TCC*, pp. 394–409, 2004.

[29] I. Haitner, J. J. Hoch, O. Reingold, and G. Segev. Finding collisions in interactive protocols – A tight lower bound on the round complexity of statistically-hiding commitments. ECCC TR07-038, 2007.

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

[31] I. Haitner and O. Reingold. A new interactive hashing theorem. *22nd CCC*, pp. 319–332, 2007.

[32] I. Haitner and O. Reingold. Statistically-hiding commitment from any one-way function. *39th STOC*, pp. 1–10, 2007.

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

[34] O. Horvitz and J. Katz. Bounds on the efficiency of "black-box" commitment schemes. *32nd ICALP*, pp. 128–139, 2005.

[35] R. Impagliazzo and S. Rudich. Limits on the provable consequences of one-way permutations. *21st STOC*, pp. 44–61, 1989.

[36] J. Kilian. Founding cryptography on oblivious transfer. *20th STOC*, pp. 20–31, 1988.

[37] J. Kilian, C. Rackoff, and E. Petrank. Lower bounds for concurrent zero knowledge. *Combinatorica*, 25(2):217–249, 2005.

[38] J. H. Kim, D. R. Simon, and P. Tetali. Limits on the efficiency of one-way permutation-based hash functions. *40th FOCS*, pp. 535–542, 1999.

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

[40] E. Kushilevitz and R. Ostrovsky. Replication is NOT needed: SINGLE database, computationally-private information retrieval. *38th FOCS*, pp. 364–373, 1997.

[41] E. Kushilevitz and R. Ostrovsky. One-way trapdoor permutations are sufficient for non-trivial single-server private information retrieval. *EUROCRYPT '00*, pp. 104–121, 2000.

[42] Y. Lindell. Parallel coin-tossing and constant-round secure two-party computation. *J. of Crypto.*, 16(3):143–184, 2003.

[43] H. Lipmaa. An oblivious transfer protocol with log-squared communication. *8th ICISC*, pp. 314–328, 2005.

[44] M. Luby. *Pseudorandomness and Cryptographic Applications*. Princeton University Press, 1996.

[45] M. Luby and C. Rackoff. How to construct pseudorandom permutations from pseudorandom functions. *SICOMP*, 17(2):373–386, 1988.

[46] M. Naor. Bit commitment using pseudorandomness. *J. of Crypto.*, 4(2):151–158, 1991.

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

[48] M. Naor and B. Pinkas. Efficient oblivious transfer protocols. *12th SODA*, pp. 448–457, 2001.

[49] M. Naor and M. Yung. Universal one-way hash functions and their cryptographic applications. *21st STOC*, pp. 33–43, 1989.

[50] M.-H. Nguyen, S. J. Ong, and S. P. Vadhan. Statistical zero-knowledge arguments for NP from any one-way function. *47th FOCS*, pp. 3–14, 2006.

[51] R. Ostrovsky and W. E. Skeith. A survey of single database PIR: Techniques and applications. ePrint 2007/059, 2007.

[52] M. O. Rabin. How to exchange secret by oblivious transfer. Technical Report TR-81, Harvard University, 1981.

[53] O. Reingold, L. Trevisan, and S. P. Vadhan. Notions of reducibility between cryptographic primitives. *1st TCC*, pp. 1–20, 2004.

[54] J. Rompel. One-way functions are necessary and sufficient for secure signatures. *22nd STOC*, pp. 387–394, 1990.

[55] A. Rosen. A note on constant-round zero-knowledge proofs for NP. *1st TCC*, pp. 191–202, 2004.

[56] S. Rudich. *Limits on the provable consequences of one-way functions*. PhD thesis, EECS Dept., UC Berkeley, 1988.

[57] D. R. Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? *EUROCRYPT '98*, pp. 334–345, 1998.

[58] H. Wee. One-way permutations, interactive hashing and statistically hiding commitments. *4th TCC*, pp. 419–433, 2007.

[59] S. Wolf and J. Wullschleger. Oblivious transfer is symmetric. *EUROCRYPT '06*, pp. 222–232, 2006.

[60] A. C. Yao. How to generate and exchange secrets. *27th FOCS*, pp. 162–167, 1986.