2016 IEEE 57th Annual Symposium on Foundations of Computer Science

Bounded-Communication Leakage Resilience via Parity-Resilient Circuits

Vipul Goyal*, Yuval Ishai†,‡, Hemanta K. Maji§, Amit Sahai‡ and Alexander A. Sherstov‡
*Microsoft Research
Bangalore, India.
Email: vipul@microsoft.com
†Department of Computer Science
Technion, Haifa, Israel.
Email: yuvali@technion.ac.il
‡Department of Computer Science
University of California, Los Angeles, USA.
Email: {sahai,sherstov}@cs.ucla.edu
§Department of Computer Science
Purdue University, West Lafayette, USA.
Email: hmaji@purdue.edu

***Abstract*—We consider the problem of distributing a computation between two parties, such that any bounded-communication leakage function applied to the local views of the two parties reveals essentially nothing about the input. This problem can be motivated by the goal of outsourcing computations on sensitive data to two servers in the cloud, where both servers can be simultaneously corrupted by viruses that have a limited communication bandwidth.**

We present a simple and efficient reduction of the above problem to that of constructing *parity-resilient circuits*, namely circuits that map an encoded input to an encoded output so that the parity of any subset of the wires is essentially independent of the input. We then construct parity-resilient circuits from circuits that are resilient to *local* leakage, which can in turn be obtained from protocols for secure multiparty computation. Our main reduction builds on a novel generalization of the "ε-biased masking lemma" that applies to interactive protocols.

Applying the above, we obtain two-party protocols with resilience to bounded-communication leakage either in the information-theoretic setting, relying on random oblivious transfer correlations, or in the computational setting, relying on non-committing encryption which can be based on a variety of standard cryptographic assumptions.

Keywords: Leakage-resilient cryptography, communication complexity, ε-biased masking.

I. Introduction

The goal of *leakage-resilient cryptography* is to maintain the traditional guarantees of cryptography even when partial information about internal secrets can be leaked. A central theme of research in this area is that of securing *general computations* against leakage. Originating from [1], [2], [3], this goal has been pursued in a variety of computational models and with different types of natural leakage functions.

In this work we focus on securing general computations against leakage in the following simple scenario. Suppose that a client wishes to outsource the computation of a function f on a sensitive input x to the cloud. The client trusts the cloud servers to perform the computation correctly, but would like to ensure that no information about x is revealed. A direct solution is to have the client encrypt x using fully homomorphic encryption (FHE) [4], and have a cloud server compute f on the encrypted input. However, FHE is currently still quite far from being practical, its existence relies on a relatively narrow class of cryptographic assumptions related to the intractability of lattice problems, and its fully compact form requires an ad-hoc circular security assumption.

A potentially more efficient alternative approach is to distribute the computation of f between two non-colluding servers. That is, the client starts by secret-sharing x between the two servers. This step should be done very efficiently, e.g., in quasi-linear time, and should be independent of the function f. Given a description of f, the servers then engage in an interactive secure two-party computation protocol [5], [6] for evaluating the shares of $y = f(x)$ from the shares of x. Finally, the servers send the shares of y back to the client, who reconstructs the output. Again, the final reconstruction step should be efficient and independent of f.

In addition to not requiring the use of FHE, such two-server solutions offer several other advantages over the single-server solution: They can minimize the amount of work performed by the client (eliminating expensive "cryptographic" computations), they can provide information-theoretic security given correlated randomness between the servers that can be set up before the input x is known, and they do not require the client to maintain a secret state for reconstructing the output. The latter feature is useful for accommodating more general scenarios in which inputs originate from and/or outputs are delivered to multiple clients.

Bounded-communication leakage. If such a two-server

0272-5428/16 $31.00 © 2016 IEEE
DOI 10.1109/FOCS.2016.10

IEEE computer society

solution is implemented using standard protocols for secure two-party computation, the client is protected against any *single* corrupted server. The question we consider is that of providing a meaningful protection even when the two servers are *simultaneously* corrupted. We envision a scenario where both servers are infected by cooperating viruses, but the viruses are subject to a bound on the total number of bits they can communicate with each other. We refer to this type of leakage as *bounded-communication leakage* (BCL). Our goal is to design BCL-resilient two-party protocols, namely ones that allow the servers to interactively compute shares of $f(x)$ from shares of x while completely hiding x from the viruses. See Figure 1 for an illustration of this motivating scenario.

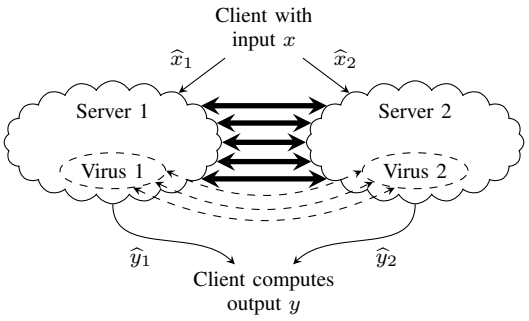

Figure 1. Motivating Example.

Note that the standard notion of security for two-party protocols coincides with BCL-resilience when the communication bound is 0 (i.e., the viruses cannot communicate at all). However, the security guarantees of standard protocols for secure two-party computation break down if even a small amount of leakage is allowed [7]. In particular, both "GMW-style" protocols and "Yao-style" protocols effectively generate a simple secret-sharing of all intermediate values of the computation, and any such intermediate value can be recovered using a single bit of leakage regardless of the amount of communication required to compute this value given the inputs alone.

Other than the restriction on the total amount of communication between the viruses, we do not impose any other restriction on the way they may interact. For instance, the viruses may first store their local views of the entire protocol execution, and only then run a multi-round (bounded-communication) protocol for recovering information about the secret input x from their local views. We do assume, however, that the viruses are *passive* in the sense that they do not tamper with the messages sent by the infected servers during the protocol execution.

The BCL assumption can be justified by the existence of virus detection mechanisms that make it difficult for the viruses to directly communicate with each other or to alter the messages sent by the servers without being detected. Still, the viruses can communicate at a slow rate, e.g., by carefully controlling the timing of the messages. Earlier works that impose bounds on the communication of adversarial parties include [8], [9], [10], [11].

A very different motivating scenario for the BCL assumption is that of protecting the computation performed by the servers against unintentional leakage of partial information resulting from the computation process itself. This type of leakage is captured by the "only computation leaks" (OCL) assumption put forward in the influential work of Micali and Reyzin [2]. The OCL assumption is typically motivated by side-channel leakage in hardware implementations, where the servers correspond to different hardware components. The BCL assumption we consider is less restrictive in that it can apply globally to the entire views of the servers throughout the protocol execution, regardless of the way in which computations are carried out. Making the stronger OCL assumption does not seem to make the problem considerably easier. On the other hand, unlike previous works on the OCL model, in this work we focus on a simple *single-execution* setting, where only one (stateless) computation is being performed, as opposed to the more challenging *continuous leakage* setting in which a sequence of computations with a common secret state are subject to leakage. The following comparison captures the best adaptations of previous results to our simpler model, ignoring FHE-based solutions [12], [13] that are trivialized in our single-execution setting.

The first goal of the present work is to study the feasibility of BCL-resilient computation.

> *Under which cryptographic assumptions or setup assumptions can general computations be protected against bounded-communication leakage?*

The prior state of the art can be summarized as follows. In the information-theoretic setting, a construction of Dziembowski and Faust [14] (the "DF-construction" for short) provides unconditional security by employing leak-free hardware components whose size must inherently grow with both the leakage bound and the statistical security parameter. Concretely, each hardware component samples a random pair of orthogonal vectors that are distributed to the two servers. The security of the construction breaks down if the entire output of any component is leaked. While some form of setup seems necessary in the information-theoretic setting even without leakage,[1] the DF-construction leaves open the possibility of using *constant-size* (or "finite") hardware components, namely ones whose size does not depend on the leakage bound or the statistical security parameter.

[1]Indeed, all known approaches for information-theoretic secure two-party computation using correlated randomness require the entropy of the correlated randomness to be bigger than the circuit size, except for the extreme case of exponential-size circuits [15]. Thus, the small amount of correlated randomness provided by the client's messages is unlikely to be sufficient.

The breakthrough work of Goldwasser and Rothblum [16] and subsequent variants of Bitansky et al. [17] show that information-theoretic OCL and BCL security is possible even without any setup. However, these protocols require a large number of servers, whereas in this work we focus on 2-server solutions.

In the computational security model, Dachman-Soled et al. [18] showed how to instantiate the hardware components of the DF-construction in the plain model by using a strong form of deniable encryption [19], whose only known instantiations rely on indistinguishability obfuscation (iO) [20], [21]. The possibility of computational solutions in the plain model that do not rely on FHE or iO remained open.

In addition to the feasibility questions, we will be interested in the achievable *leakage rate*, measured as the ratio between the leakage communication bound c and the size S of the circuits required for implementing the protocol for f. (This captures the fractional information leakage about internal computation steps.) We will also be interested in the *computational overhead* of protocols, measured as the ratio between S and the circuit size of f, denoted by s.

> *What are the best achievable leakage rate and computational overhead of BCL-resilient protocols?*

In previous two-party solutions that do not rely on FHE the leakage rate is worse than $1/cs$, which is inherited from the parameters of the DF-construction (see Table 1 in [18]). Moreover, the computational overhead of these solutions is at least quadratic in c. The latter holds also for protocols from [16], [17] that involve more than two servers.

A. Our Results

We introduce a simple and general technique for constructing and analyzing BCL-resilient protocols. Our technique yields efficient protocols that achieve information-theoretic security using a minimal setup, or alternatively provide computational security under a variety of standard assumptions.

The BCL-resilient protocols we construct can be obtained from any standard protocol for secure multiparty computation (MPC), where the security threshold of the MPC protocol serves as a leakage communication bound in the BCL-resilient protocol. Alternatively, they can be obtained from any *multi-server* OCL-resilient protocol (such as the one from [16]), where the statistical error of the multi-server protocol determines the leakage bound of the two-server protocol.

More concretely, we obtain the following new results on BCL-resilient protocols.

- FEASIBILITY: INFORMATION-THEORETIC SETTING. We construct (2-server) BCL-resilient protocols with information-theoretic security using only an oblivious transfer (OT) setup. That is, the servers either have access to an OT oracle or to constant-size leak-free hardware components that produce OT correlations.[2] As discussed above, this is the best one can hope for barring a major breakthrough in information-theoretic cryptography. This should be contrasted with the hardware components required by the DF-construction, whose size should grow with both the leakage bound and the security parameter.
- FEASIBILITY: COMPUTATIONAL SETTING. We obtain the first 2-server computational BCL-resilient protocols (in the plain model) that do not rely on FHE or iO. Concretely, our construction only requires the use of *non-committing encryption* (NCE) [22], which can be based on a variety of standard cryptographic assumptions that include the intractability of factoring Blum integers [23] or Decisional Diffie Hellman [24]. These instantiations make a crucial use of the simple setup of our information-theoretic protocols.
- LEAKAGE RATE AND COMPUTATIONAL OVERHEAD. Our BCL-resilient protocols, in both settings, offer qualitative improvements in leakage rate and efficiency over previous protocols. Recall that in the information-theoretic setting, the 2-server DF-construction [14] the leakage bound is inherently smaller than the size of the trusted hardware compontents. Moreover, previous solutions in both settings [14], [16], [17], [18] involve multiplicative computational overhead on the server side which is bigger than the leakage bound. Our protocols get around these limitations. In particular, whenever f can be computed by a circuit whose size s and depth h satisfy the mild conditions $s \geq c^2$ and $s \geq h^2$, where c is the leakage communication bound, we can tolerate $\tilde{\Omega}(\sqrt{s})$ bits of leakage with $\log^{O(1)}(c)$ computational overhead using either constant-size hardware (alternatively, OT-correlations) or NCE.
  More generally, if $f\colon \{0,1\}^n \to \{0,1\}^n$ is computable by a circuit of size s and depth h, then we get BCL-resilient protocols with security against c bits of leakage where the client can be implemented by circuits of size $\widetilde{O}(n+c)$, and where the servers can be implemented by circuits of size $\widetilde{O}(s+ch+c^2)$ in the information-theoretic protocols, or alternatively invoke an NCE protocol a similar number of times in the computational protocols. These parameters are inherited from known honest-majority MPC protocols [25]. The information-theoretic protocols can be efficiently implemented by having the client distribute the instances of the OT correlation to the servers in an offline preprocessing

[2] An OT correlation delivers a pair of random bits (s_0, s_1) to one server and the pair (b, s_b) to the other, where b is a random bit.

phase, before the input x is known.[3]

The above results are based on two technical ingredients that may be considered as independently interesting results: (1) the construction of so-called *parity-resilient circuits*, a natural computational analogue of small-bias generators [26]; and (2) a generalization of the so-called ε-*biased masking* lemma [27] that applies to interactive protocols. These are described in the next section.

B. Overview of Techniques

We start by observing what goes wrong when trying to evaluate f using a very simple protocol for secure two-party computation, namely the "GMW protocol" [6], [28] for passive adversaries when implemented over ideal OT. In this protocol, the local views of the two parties essentially form an *additive secret sharing* of the gate values in a circuit computing f. That is, letting z denote the vector of gate values on the input x, the joint views are distributed as $(z \oplus r, r)$ where r is a uniformly random bit-string.

This protocol miserably fails in achieving our goal: any intermediate value z_i in the computation of f can be revealed by leaking only a single bit from a local view. Moreover, by the $\mathbb{F}_2$-linearity of the secret sharing of z, revealing an arbitrary *parity* of bits from a local view reveals the corresponding parity of the bits of z.

The first idea is that in order to protect the protocol against such simple parity attacks, it suffices to protect the computation of f via a *parity-resilient circuit*: a randomized circuit that receives a randomized encoding $\widehat{x}$ of an input x and produces an encoded output $\widehat{y}$, where the parity of any subset of the gate values (when evaluating the circuit on $\widehat{x}$) reveals essentially nothing about x. A bit more precisely, the circuit is ε-parity-resilient if for any inputs x, x', the distributions of the gate values $z(\widehat{x})$ and $z(\widehat{x'})$ are ε-indistinguishable by parity functions.

Using such a parity-resilient circuit, we can easily obtain a protocol with resilience to a *single bit of parity leakage*. The client locally encodes the input x, and additively shares the encoding $\widehat{x}$ between the two servers. The servers now run the GMW protocol to obtain additive shares of an output encoding $\widehat{y}$. The shares of $\widehat{y}$ are sent back to the client, who decodes y.

Constructing parity-resilient circuits. We turn to the question of constructing parity-resilient circuits. Despite this being a very natural object, we are not aware of any previous related study. Our first observation is that any multi-party protocol which is ε-resilient to a single bit of OCL leakage per computation step, such as the protocol of [16], directly implies an ε-parity-resilient circuit. Indeed, arbitrarily representing each computation step by a boolean circuit, the parity of an arbitrary subset of gates can be recovered using a single bit of leakage from each computation step. However, given the complexity and the relatively poor parameters of the construction from [16] and its variants, we seek an alternative and more direct construction.

Our main construction of parity-resilient circuits can be based on any general MPC protocol that offers security against k passively corrupted parties. The construction is broken into several modular steps, where the first two steps mimic previous constructions of small-bias PRGs from bounded independence [29], [30]. The first step is a transformation of a k-secure MPC protocol into a k-private circuit, namely a circuit with the same syntax as parity-resilient circuits, except that the joint values of any k gates should reveal nothing about x. The transformation, pointed out in [1], is straightforward: let f' denote a randomized function which maps a secret-shared input for f to a secret-shared output for f. Then, a k-secure MPC protocol for f' can be directly implemented as a k-private circuit. The second step replaces each atomic gate in the k-private circuit by a constant-size ("finite") randomized gadget that maps a non-linear encoding of the inputs to a non-linear encoding of the output. The existence of a gadget with the required properties follows by a probabilistic argument (cf. [31]), however we also give a simple explicit construction of a gadget $g : \{0,1\}^9 \to \{0,1\}^3$.

The combination of the above two steps gives a construction of $2^{-\Omega(k)}$-parity-resilient circuits over the finite gadget, where the size of a parity resilient-circuit for f is linear in the circuit size of the k-secure MPC protocol for f'. Somewhat surprisingly, turning a circuit over g to a circuit over a standard binary gates is not as straightforward as it may seem. In particular, a naive implementation of g will compromise parity resilience. Instead, we will consider for now a natural generalization of the GMW protocol that works over g-gates instead of binary gates by using a finite two-party oracle H instead of the standard OT oracle. When applied to a parity-resilient circuit over g, this protocol generates a pair of views distributed as $(z \oplus r, r)$, where z is a parity-resilient encoding of the input. The protocol still offers resilience to a single bit of parity-leakage.

From parities to bounded communication via generalized ε-biased masking. Our next, and most technical, step is arguing that parity-resilience *automatically* implies general BCL-resilience with related parameters. Concretely, we prove the following theorem. Suppose that μ_0 and μ_1 are two distributions that are ε-indistinguishable by parities. Then any interactive two-party protocol with c bits of communication can have at most an $\varepsilon \cdot 2^{c/2}$ advantage in distinguishing between a secret sharing of μ_0 and μ_1, namely between the distributions $(\mu_0 \oplus r, r)$ and $(\mu_1 \oplus r, r)$, where r is a random bit-string chosen independently of μ_0, μ_1. This means that as long as $\varepsilon \ll 2^{-c/2}$, an ε-parity-resilient

[3] Note that the DF-construction can also be implemented in this setting by having the client distribute (large) instances of an inner product correlation instead of OT correlation. However, our protocol has the efficiency advantages discussed above.

circuit (combined with the GMW protocol) offers good BCL-resilience against c bits of leakage. Together with the above, we get a transformation from k-secure MPC protocols to BCL-resilient protocols over a finite two-party oracle H that tolerate $\Omega(k)$ bits of leakage.

Specializing the above theorem to 1-message protocols and for the case where one of the two distributions is uniform, the theorem can be shown to be essentially equivalent to the so-called *ε-biased masking* lemma from [27]. The ε-biased masking lemma says that $M \oplus X$, where M is a high entropy source and S is an independent small-bias distribution, is statistically close to uniform. See the full version [40] for a proof of the equivalence. Our theorem is more general in two orthogonal ways: it applies to multi-round protocols, and it extends pseudorandomness to indistinguishability. In light of the usefulness of the original ε-biased masking lemma in cryptography (see, e.g., [32], [33], [34], [30]) we expect our generalized version to find additional applications.

Wrapping up. The above is almost enough to get our results for the information-theoretic model. The only remaining step is to replace the finite oracle H by a standard OT oracle, or alternatively an OT correlation. (The latter protocols directly imply parity-resilient circuits over the binary basis.) This step follows by observing that a simple deterministic reduction from H to OT respects BCL-resilience. More generally, to respect BCL-resilience we need such reductions to satisfy a strong notion of security we refer to as *joint simulation security*. This notion, previously considered in [18], strengthens the standard simulation-based definition of secure computation by considering the outputs of the two simulators *jointly*. To make this possible, the two simulators share a common source of randomness. Finally, we obtain our results for the computational model by observing that the standard construction of OT from NCE is also secure under the strong joint simulation requirement.

These final steps crucially rely on the constant-size setup feature of our information-theoretic protocols. Indeed, the setup required previous DF-construction could only be instantiated in the plain model using iO [18]. This qualitative difference between our constant-size setup and the computationally simple inner-product setup required by DF may seem surprising. However, natural attempts to realize the DF setup via standard protocols for secure two-party computation (even ones that offer adaptive security [22]) fail. This can be attributed to the fact, already discussed above, that applying a low-communication leakage attack to standard secure computation protocols [5], [6] reveals intermediate values of the computation that cannot be computed via a low-communication protocol.

More broadly, obtaining "leakage tolerant" forms of information-theoretic protocols for functions with a super-polynomial input domain appears to be a difficult task even for simple functions [35]. We bypass this problem by using constant-size oracles, whose brute-force secure evaluation using a truth-table representation is trivially leakage tolerant.

See Figure 2 for a roadmap of the different steps of our construction and the relations between different types of leakage-resilient objects and Section II for the relevant definitions and formal theorem statements.

C. Open Problems

Our work gives rise to two natural open questions: extending the results from the single-execution setting to the more challenging continuous leakage setting, and obtaining similar information-theoretic results in a multi-party setting without a setup (reproducing the result of [16] with better parameters). We believe that the ideas introduced in this work will serve as a useful basis for such extensions.

II. Outline of Definitions and Theorems

A. Definitions

We consider boolean circuits over a basis $\mathbb{B}$ of gates. Each gate in $\mathbb{B}$ computes a function of the form $g : \{0,1\}^{\alpha} \to \{0,1\}^{\beta}$. When the basis is not specified, it is understood to be the full binary basis $\mathbb{B} = \{\mathsf{AND}, \mathsf{OR}, \mathsf{NOT}, \mathsf{XOR}\}$ where $\mathsf{AND}, \mathsf{OR}, \mathsf{XOR}$ gates have fan-in 2. We will consider both deterministic and randomized circuits (a circuit is deterministic by default). We let $|C|$ denote the number of gates in C, also including inputs and randomness gates.

We consider a simple model for leakage-resilient circuits that generalizes the stateless variant of private circuits from [1] (see also [3], [36]). Such circuits map an encoded input for a function f into an encoded output, where the internal gate values of C should hide the input in the presence of partial leakage.

Definition 1 (Leakage-resilient circuits and k-private circuits)**.** *For $f\colon \{0,1\}^{n_i} \to \{0,1\}^{n_o}$, a leakage-resilient circuit for f is defined by (I, C, O), where:*

- *$I\colon \{0,1\}^{n_i} \to \{0,1\}^{\widehat{n}_i}$ is a randomized* input encoder *circuit, which maps an input x to an encoded input $\widehat{x}$;*
- *C is a randomized circuit, mapping an encoded input $\widehat{x} \in \{0,1\}^{\widehat{n}_i}$ to an encoded output $\widehat{y} \in \{0,1\}^{\widehat{n}_o}$;*
- *$O\colon \{0,1\}^{\widehat{n}_o} \to \{0,1\}^{n_o}$ is a deterministic* output decoder *circuit, which maps an encoded output $\widehat{y}$ to an output y.*

*We say that (I, C, O) is an (L, ε)-*leakage-resilient implementation of *f, for a leakage function $L\colon \{0,1\}^{|C|} \to \{0,1\}^*$ and $\varepsilon \geq 0$, if the following requirements hold:*

- Correctness*: For any input $x \in \{0,1\}^{n_i}$, we have $\Pr[O(C(I(x))) = f(x)] = 1$, where the probability is over the randomness of I and C;*
- Leakage-resilience*: For any $x, x' \in \{0,1\}^{n_i}$, the statistical distance between the distributions $L(C[I(x)])$ and $L(C[I(x')])$ is at most ε, where $C[\widehat{x}]$ denotes the joint*

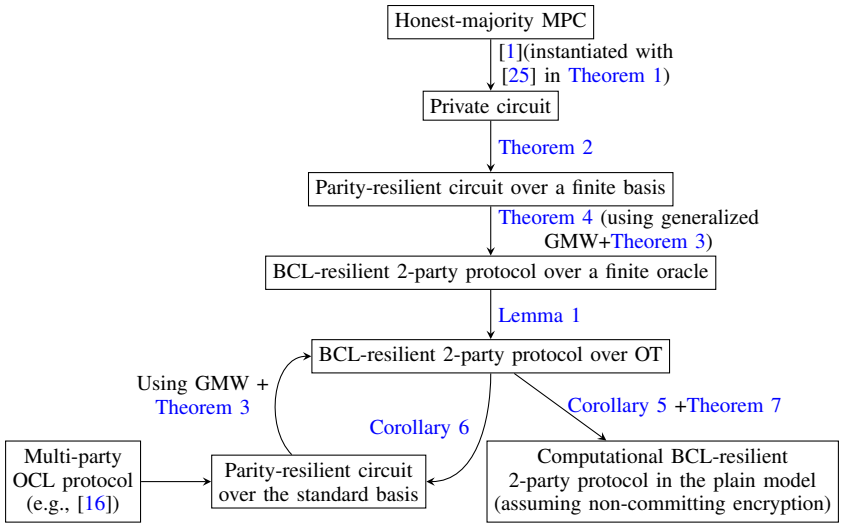

Figure 2. Relations between different notions of leakage-resilient circuits and protocols.

distribution of the $|C|$ gate values in the computation of C on input $\widehat{x}$.

For a class $\mathcal{L}$ of leakage functions, we say that (I, C, O) is an $(\mathcal{L}, \varepsilon)$-leakage-resilient implementation of f, *if it is an (L, ε)-leakage-resilient implementation of f for all $L \in \mathcal{L}$. We say that (I, C, O) is a* k-private implementation of f *if it is an $(\mathcal{L}, 0)$-leakage-resilient implementation of f for the class $\mathcal{L}$ of all k-bit projection functions (which output k fixed entries of the input).*

Without any requirements on I and O, the above definition can be met by having I compute a leakage-resilient secret sharing of the input that is passed by C directly to the output decoder. The decoder decodes the circuit output and computes f. To rule out such a solution, we require the encoder and the decoder to be *universal* (i.e., depend only on n_i, n_o and the circuit size of f but not on f itself). Furthermore, we would like the encoder and decoder size to be considerably smaller than the circuit size f. These requirements effectively force C to perform the bulk of the computation in a leakage-resilient manner.

The following bound on the efficiency of k-private circuits follows by implementing a secure multiparty computation protocol from [25] as a circuit, via the general transformation suggested in [1]. The protocol achieves security against k semi-honest parties by using $n = O(k)$ parties. The private circuit is obtained by applying the protocol to the function which maps a secret sharing of the input for f to a secret sharing of the output of f. In the private circuits model, the secret sharing of the input is implemented by the input encoder and the reconstruction of the output from its shares is implemented by the output decoder.

Theorem 1 (Implicit in [25])**.** *There is an efficient algorithm Q such that for every positive integer k and every circuit C_f of size s and depth h computing a function $f\colon \{0,1\}^{n_i} \to \{0,1\}^{n_o}$, the output of $Q(1^k, C_f)$ is a k-private implementation (I, C, O) of f with $|I| = \widetilde{O}(n_i + k)$, $|C| = \widetilde{O}(s + kh + k^2)$ and $|O| = \widetilde{O}(n_o + k)$.*

We will be particularly interested in the following special case of leakage-resilient circuits.

Definition 2 (Parity-resilient circuits)**.** *We say that (I, C, O) is an* ε-parity-resilient implementation of f *if it is an $(\mathcal{L}, \varepsilon)$-leakage-resilient implementation of f for the class $\mathcal{L}$ of all parity functions, namely the class of functions that output the parity of a subset of the wires.*

The main goal of this work is to construct two-party protocols that are resilient to bounded-communication leakage. We consider two-party protocols that start with encoded inputs and end with encoded outputs. Furthermore, we also consider protocols that receive correlated random inputs, or alternatively invoke a two-argument function as an oracle.

Definition 3 (Two-party protocol with encoded input and output)**.** *A two-party protocol for $f\colon \{0,1\}^{n_i} \to \{0,1\}^{n_o}$ is defined by $\Pi = (I, (R_1, R_2), (M_1, M_2), O)$, where:*

- *$I\colon \{0,1\}^{n_i} \to \{0,1\}^{\widehat{n}_i} \times \{0,1\}^{\widehat{n}_i}$ is a randomized* input encoder *circuit, which maps an input x for f to a pair of protocol inputs $(\widehat{x}_1, \widehat{x}_2)$.*
- *R_1 and R_2 are distributions over $\{0,1\}^{n_r}$ that capture the random inputs of the two parties. They are assumed to be uniform and independent by default. Otherwise, we say that Π uses* correlated randomness *(R_1, R_2).*
- *M_1 and M_2 are deterministic* next message functions*, where M_j determines the next message to be sent by party j as a function of its input $\widehat{x}_j$, random input r_j,*

and the sequence of previous messages received from the other party. Messages are sent in rounds, where party 1 sends messages in odd rounds and party 2 in even rounds. After a predetermined number of rounds, the function M_j returns a local output $\widehat{y}_j \in \{0,1\}^{\widehat{n}_o}$ for party j.

- $O\colon \{0,1\}^{\widehat{n}_o} \times \{0,1\}^{\widehat{n}_o} \to \{0,1\}^{n_o}$ *is a deterministic* output decoder *circuit, which maps a pair of protocol outputs $(\widehat{y}_1, \widehat{y}_2)$ to an output y of f.*

For $x \in \{0,1\}^{n_i}$, we denote by $\Pi(x)$ the output of π on input x, namely the result of applying the input encoder I to x, interacting as specified by $(R_1, R_2), (M_1, M_2)$, and applying the output decoder O to the pair of protocol outputs. We say that Π correctly *computes $f\colon \{0,1\}^{n_i} \to \{0,1\}^{n_o}$ if for every input $x \in \{0,1\}^{n_i}$ we have $\Pr[\Pi(x) = f(x)] = 1$.*

We denote by $\mathsf{view}(x)$ the joint distribution $(\mathsf{view}_1, \mathsf{view}_2)$ obtained by running Π on input x, where view_j includes the encoded input $\widehat{x}_j$, the random input r_j (sampled from R_j), and the sequence of messages received *by party j. (The messages sent by party j as well as its output $\widehat{y}_j$ are uniquely determined by view_j.) We denote by $|\Pi|$ the total circuit size required for implementing all invocations of the next message functions, including the input and randomness gates.*

Finally, we also consider oracle-aided protocols*, where the parties can invoke a two-party oracle $H : \{0,1\}^{\alpha_1} \times \{0,1\}^{\alpha_2} \to \{0,1\}^{\beta}$. After receiving an input from each party, the oracle delivers the output to party 2. In an oracle-aided protocol, the outputs of the next message function M_j also include messages sent by party j as inputs to H and the inputs of M_2 include messages received by party 2 as outputs of H. The oracle may be invoked several times in parallel, where each party can send inputs to several invocations of H in a single round.*

We now define our main notion of bounded-communication leakage resilience.

Definition 4 (BCL-resilient protocol)**.** *We say that Π is a (c, ε)-*bounded-communication leakage resilient protocol for f *(or (c, ε)-BCL-resilient for short) if Π correctly computes f, and the following security requirement holds. For any communication protocol $\pi : \{0,1\}^{n_1} \times \{0,1\}^{n_2} \to \{0,1\}$ with communication complexity at most c, and any pair of inputs $x, x' \in \{0,1\}^{n_i}$ we have $|\Pr[\pi(\mathsf{view}(x)) = 1] - \Pr[\pi(\mathsf{view}(x')) = 1]| \leq \varepsilon$, where $n_j = |\mathsf{view}_j|$.*

For a polynomial-time computable $f : \{0,1\}^* \to \{0,1\}^*$, the above definition can be naturally extended to capture *computational* BCL-resilient protocols for f. Such a protocol is specified by PPT algorithms $\Pi = (I, (M_1, M_2), O)$. We say that Π is computationally $c(n)$-BCL-resilient if for every input length n it is $(c(n), \varepsilon(n))$-BCL-resilient for some negligible function ε, with respect to every leakage protocol π that is implemented by circuits of size $\mathsf{poly}(n)$.

B. Main Theorems

Following are the main theorems and corollaries that correspond to the steps of the construction described in Section I-B. See Figure 2 for a roadmap. Transformations between different objects are always polynomial-time computable, even when we do not say so explicitly.

Theorem 2 (Private circuits ⇒ parity-resilient circuits over a finite basis)**.** *There is a function $g\colon \{0,1\}^9 \to \{0,1\}^3$ such that every k-private implementation (I, C, O) of a function f can be efficiently transformed into a $2^{-\Omega(k)}$-parity-resilient implementation (I', C', O') of f over the basis $\mathbb{B} = \{g\}$, with $|I'| = O(|I|)$, $|C'| = O(|C|)$, and $|O'| = O(|O|)$.*

Theorem 3 (Generalized ε-biased masking)**.** *Let μ_0, μ_1 be probability distributions over $\{0,1\}^n$ that are ε-indistinguishable by parities. Then any communication protocol $\pi : \{0,1\}^n \times \{0,1\}^n \mapsto \{0,1\}$ with communication complexity at most c obeys:*

$$\left| \mathop{\mathbb{E}}_{x \sim \mu_0} \mathop{\mathbb{E}}_{r \xleftarrow{\$} \{0,1\}^n} [\pi(x \oplus r, r)] - \mathop{\mathbb{E}}_{x \sim \mu_1} \mathop{\mathbb{E}}_{r \xleftarrow{\$} \{0,1\}^n} [\pi(x \oplus r, r)] \right| \leq 2^{c/2}\varepsilon$$

Theorem 4 (Parity-resilient circuits ⇒ BCL-resilient oracle-aided protocols)**.** *Suppose (I', C', O') is a 2^{-k}-parity-resilient implementation of f over a basis $\mathbb{B} = \{g\}$, where $g : \{0,1\}^{\alpha} \to \{0,1\}^{\beta}$, and where C' has depth h. Then there is a (c, ε)-BCL-resilient $O(h)$-round two-party protocol $\Pi = (I'', (R_1, R_2), (M_1, M_2), O'')$ for f using an oracle $H : \{0,1\}^{\alpha+\beta} \times \{0,1\}^{\alpha} \to \{0,1\}^{\beta}$, with $c = \Omega(k)$, $\varepsilon = 2^{-\Omega(k)}$, $|I''| = O(|I'|)$, $|R_1| = \beta \cdot |C'|$, $|R_2| = 0$, $|O''| = O(|O'|)$, and $|\Pi| = O(|C'|)$.*

Lemma 1 (Finite oracle ⇒ OT oracle)**.** *For any positive integers $\alpha_1, \alpha_2, \beta$, a (c, ε)-BCL-resilient H-aided protocol Π for f, where $H : \{0,1\}^{\alpha_1} \times \{0,1\}^{\alpha_2} \to \{0,1\}^{\beta}$, can be efficiently transformed to a similar OT-aided protocol Π' for f, where Π' has the same input encoder and output decoder as Π, and where $|\Pi|' \leq 2^{\alpha_2} \cdot \beta \cdot |\Pi|$.*

Corollary 5 (BCL-resilient protocols over OT)**.** *For every positive integer k and circuit C_f of size s and depth h computing a function $f\colon \{0,1\}^{n_i} \to \{0,1\}^{n_o}$, there is a $(k, 2^{-k})$-BCL-resilient OT-aided protocol $\Pi = (I, (R_1, R_2), (M_1, M_2), O)$ for f, where $|\Pi| = \widetilde{O}(s + kh + k^2)$, $|I| = \widetilde{O}(n_i + k)$, and $|O| = \widetilde{O}(n_o + k)$, and where the OT oracle is called $\widetilde{O}(s + kh + k^2)$ times. Alternatively, there is a similar protocol that uses independent instances of OT correlation instead of an OT oracle.*

Corollary 6 (Parity-resilient circuits over a binary basis)**.** *For every positive integer k and circuit C_f of size s and depth h computing a function $f\colon \{0,1\}^{n_i} \to \{0,1\}^{n_o}$, there*

is a 2^{-k}-parity-resilient implementation (I, C, O) of f over the full binary basis, with $|I| = \widetilde{O}(n_i + k)$, $|C| = \widetilde{O}(s + kh + k^2)$, and $|O| = \widetilde{O}(n_o + k)$.

Theorem 7 (Computational BCL-resilient protocols in the plain model)**.** *Suppose the DDH assumption holds. Then, for every polynomial-time computable $f : \{0,1\}^n \to \{0,1\}^{m(n)}$ and polynomial $c(n)$, there is a computational $c(n)$-BCL-resilient implementation $\Pi = (I, (M_1, M_2), O)$ of f, where the running time of I is $\widetilde{O}(n + c(n))$ and the running time of O is $\widetilde{O}(m(n) + c(n))$.*

III. Communication Complexity Bound

In this section, we present a generalization of the popular "Small-bias Masking Lemma" [26], [37], [38], [27] (refer to the full version [40]) in the two-party setting.

Theorem 3 Restated (Generalized ε-biased masking)**.** *Let μ_0, μ_1 be probability distributions over $\{0,1\}^n$ that are ε-indistinguishable by parities. Then any communication protocol $\pi : \{0,1\}^n \times \{0,1\}^n \mapsto \{0,1\}$ with communication complexity at most c obeys:*

$$\left| \mathop{\mathbb{E}}_{x\sim\mu_0} \mathop{\mathbb{E}}_{r\xleftarrow{\$}\{0,1\}^n} [\pi(x \oplus r, r)] - \mathop{\mathbb{E}}_{x\sim\mu_1} \mathop{\mathbb{E}}_{r\xleftarrow{\$}\{0,1\}^n} [\pi(x \oplus r, r)] \right| \leq 2^{c/2}\varepsilon$$

Intuitively, the theorem states the following. Suppose we have a distribution μ_0 and μ_1 that are ε-indistinguishable from each other w.r.t. any linear test. That is, given a sample that is drawn according to the distribution μ_b (where $b \xleftarrow{\$} \{0,1\}$), any linear test can predict b with at most ε advantage.

Now, we additively secret share a sample according to the distribution μ_b, for $b \xleftarrow{\$} \{0,1\}$, among two parties, i.e. the joint views of parties is $(U, \mu_b \oplus U)$. Next, the parties run a low communication protocol (communication complexity bounded by c) among themselves. Now, given this communication, the advantage to predict b is at most $2^{c/2}\varepsilon$, i.e. at most $2^{c/2}$ times the advantage to predict b using linear tests on a sample drawn according to μ_b.

To prove this result, we will need some elementary Fourier analysis. The characters $\chi_S \colon \{0,1\}^n \to \mathbb{R}$ of the Fourier transform are given by $\chi_S(x) = (-1)^{S \cdot x}$. The Fourier coefficients of a function $f \colon \{0,1\}^n \to \mathbb{R}$ are denoted by: $\widehat{f}(S) = \frac{1}{N} \sum_{x\in\{0,1\}^n} f(x)\chi_S(x)$, where $N = 2^n$.

Definition 5 (ε-Indistinguishability by Parities)**.** *Two probability distributions μ_0, μ_1 over $\{0,1\}^n$ are called* ε-indistinguishable by parities *if, for every $S \subseteq [n]$,*

$$\left| \sum_{x\in\{0,1\}^n} \mu_0(x)\chi_S(x) - \sum_{x\in\{0,1\}^n} \mu_1(x)\chi_S(x) \right| \leq \varepsilon$$

Equivalently, for every $S \subseteq [n]$,

$$\left| \Pr_{x\sim\mu_0}[\chi_S(x) = 1] - \Pr_{x\sim\mu_1}[\chi_S(x) = 1] \right| \leq \varepsilon$$

We need to show:

$$\left| \mathop{\mathbb{E}}_{x\sim\mu_0} \mathop{\mathbb{E}}_{r\xleftarrow{\$}\{0,1\}^n} [\pi(x \oplus r, r)] - \mathop{\mathbb{E}}_{x\sim\mu_1} \mathop{\mathbb{E}}_{r\xleftarrow{\$}\{0,1\}^n} [\pi(x \oplus r, r)] \right| \leq 2^{c/2}\varepsilon \quad (1)$$

Let Δ stand for the left-hand side of Equation 1. Consider the matrix:

$$P = 2^{-n} \left[\mu_0(x \oplus r) - \mu_1(x \oplus r)\right]_{x,r\in\{0,1\}^n}$$

We have

$$\begin{aligned}
\Delta &= \left| \left(2^{-n} \sum_{x,r\in\{0,1\}^n} \mu_0(x)\pi(x \oplus r, r) \right) \right. \\
&\quad \left. - \left(2^{-n} \sum_{x,r\in\{0,1\}^n} \mu_1(x)\pi(x \oplus r, r) \right) \right| \\
&= \left| 2^{-n} \sum_{x,r\in\{0,1\}^n} (\mu_0(x \oplus r) - \mu_1(x \oplus r)) \cdot \pi(x, r) \right| \\
&= |\langle P, \pi \rangle|
\end{aligned}$$

where we view π as a matrix $\pi = [\pi(x, r)]_{x,r}$. Decompose the protocol as a sum of combinatorial rectangles:

$$\pi = \sum_{i\in[2^c]} \mathbf{1}_{A_i} \mathbf{1}_{B_i}^{\mathsf{T}},$$

where $A_i, B_i \subseteq \{0,1\}^n$ are some sets and $\mathbf{1}_{A_i}$, $\mathbf{1}_{B_i}$ are their, respective, characteristic vectors. Then,

$$\begin{aligned}
\Delta &= |\langle P, \pi \rangle| \\
&\leq \sum_{i\in[2^c]} \left| \mathbf{1}_{A_i}^{\mathsf{T}} P \mathbf{1}_{B_i} \right| \\
&\leq \sum_{i\in[2^c]} \sqrt{|A_i| \cdot |B_i|}\, \|P\| \\
&\leq 2^{c/2} \sqrt{\sum_{i\in[2^c]} |A_i| \cdot |B_i|}\, \|P\| \\
&\leq 2^{c/2} 2^n \|P\| \\
&= 2^{n+c/2} \|P\|,
\end{aligned}$$

where $\|\cdot\|$ denotes the spectral norm (equivalently, the largest singular value).

Claim 1. $\|P\| \leq \varepsilon 2^{-n}$

Proof: The singular value decomposition of P is found as follows:

$$P = \left[\sum_{S\subseteq[n]} 2^{-n}\left(\widehat{\mu_0}(S) - \widehat{\mu_1}(S)\right)\cdot\chi_x(S)\chi_r(S)\right]_{x,r}$$
$$= \left[\chi_S(x)\right]_{x,S}\begin{bmatrix}\ddots & & \\ & 2^{-n}\left(\widehat{\mu_0}(S) - \widehat{\mu_1}(S)\right) & \\ & & \ddots\end{bmatrix}\left[\chi_S(r)\right]_{S,r}$$
$$= H\cdot\begin{bmatrix}\ddots & & \\ & 2^{-n}\left(\widehat{\mu_0}(S) - \widehat{\mu_1}(S)\right) & \\ & & \ddots\end{bmatrix}\cdot H^{\mathsf{T}},$$

where $H = 2^{-n/2}\left[\chi_S(x)\right]_{x,S}$ is a unitary matrix. In particular,

$$\|P\| = \max_{S\subseteq[n]}\left|\widehat{\mu_0}(S) - \widehat{\mu_1}(S)\right| \leq \varepsilon 2^{-n},$$

where the final step uses the assumption that μ_0, μ_1 are ε-indistinguishable. ∎

Thus, we get that $\Delta \leq 2^{n+c/2}\|P\| \leq 2^{c/2}\varepsilon$.

For the missing details, please refer to the full version of this paper [40].

Acknowledgements: This research was done in part while the first four authors were visiting the Simons Institute for the Theory of Computing, supported by the Simons Foundation and by the DIMACS/Simons Collaboration in Cryptography through NSF grant #CNS-15-23467. The individual authors were supported by the following grants: BSF grant 2012378, ISF grant 1709/14, ERC starting grant 259426, NSF grant CNS-1566499, a DARPA/ARL SAFEWARE award, NSF Frontier Award 1413955, NSF grants 12-28984, 11-36174, 11-18096, and 10-65276, NSF CAREER award CCF-1149018, a Xerox Faculty Research Award, a Google Faculty Research Award, an equipment grant from Intel, an Okawa Foundation Research Grant, and an Alfred P. Sloan Foundation Research Fellowship. This material is in part based upon work supported by the Defense Advanced Research Projects Agency through the ARL under Contract W911NF-15-C-0205. The views expressed are those of the authors and do not reflect the official policy or position of the Department of Defense, the National Science Foundation, or the U.S. Government.

REFERENCES

[1] Y. Ishai, A. Sahai, and D. Wagner, "Private circuits: Securing hardware against probing attacks," in *CRYPTO 2003*, ser. LNCS, D. Boneh, Ed., vol. 2729. Springer, Heidelberg, Aug. 2003, pp. 463–481.

[2] S. Micali and L. Reyzin, "Physically observable cryptography (extended abstract)," in *TCC 2004*, ser. LNCS, M. Naor, Ed., vol. 2951. Springer, Heidelberg, Feb. 2004, pp. 278–296.

[3] S. Faust, T. Rabin, L. Reyzin, E. Tromer, and V. Vaikuntanathan, "Protecting circuits from leakage: the computationally-bounded and noisy cases," in *EUROCRYPT 2010*, ser. LNCS, H. Gilbert, Ed., vol. 6110. Springer, Heidelberg, May 2010, pp. 135–156.

[4] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *41st ACM STOC*, M. Mitzenmacher, Ed. ACM Press, May / Jun. 2009, pp. 169–178.

[5] A. C.-C. Yao, "Protocols for secure computations (extended abstract)," in *23rd FOCS*. IEEE Computer Society Press, Nov. 1982, pp. 160–164.

[6] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game or A completeness theorem for protocols with honest majority," in *19th ACM STOC*, A. Aho, Ed. ACM Press, May 1987, pp. 218–229.

[7] N. Bitansky, R. Canetti, and S. Halevi, "Leakage-tolerant interactive protocols," in *TCC 2012*, ser. LNCS, R. Cramer, Ed., vol. 7194. Springer, Heidelberg, Mar. 2012, pp. 266–284.

[8] S. Dziembowski, "Intrusion-resilience via the bounded-storage model," in *TCC 2006*, ser. LNCS, S. Halevi and T. Rabin, Eds., vol. 3876. Springer, Heidelberg, Mar. 2006, pp. 207–224.

[9] G. Di Crescenzo, R. J. Lipton, and S. Walfish, "Perfectly secure password protocols in the bounded retrieval model," in *TCC 2006*, ser. LNCS, S. Halevi and T. Rabin, Eds., vol. 3876. Springer, Heidelberg, Mar. 2006, pp. 225–244.

[10] S. Dziembowski and K. Pietrzak, "Intrusion-resilient secret sharing," in *48th FOCS*. IEEE Computer Society Press, Oct. 2007, pp. 227–237.

[11] I. Damgård, J. B. Nielsen, and D. Wichs, "Isolated proofs of knowledge and isolated zero knowledge," in *EUROCRYPT 2008*, ser. LNCS, N. P. Smart, Ed., vol. 4965. Springer, Heidelberg, Apr. 2008, pp. 509–526.

[12] S. Goldwasser and G. N. Rothblum, "Securing computation against continuous leakage," in *CRYPTO 2010*, ser. LNCS, T. Rabin, Ed., vol. 6223. Springer, Heidelberg, Aug. 2010, pp. 59–79.

[13] A. Juma and Y. Vahlis, "Protecting cryptographic keys against continual leakage," in *CRYPTO 2010*, ser. LNCS, T. Rabin, Ed., vol. 6223. Springer, Heidelberg, Aug. 2010, pp. 41–58.

[14] S. Dziembowski and S. Faust, "Leakage-resilient circuits without computational assumptions," in *TCC 2012*, ser. LNCS, R. Cramer, Ed., vol. 7194. Springer, Heidelberg, Mar. 2012, pp. 230–247.

[15] A. Beimel, Y. Ishai, R. Kumaresan, and E. Kushilevitz, "On the cryptographic complexity of the worst functions," in *TCC 2014*, ser. LNCS, Y. Lindell, Ed., vol. 8349. Springer, Heidelberg, Feb. 2014, pp. 317–342.

[16] S. Goldwasser and G. N. Rothblum, "How to compute in the presence of leakage," in *53rd FOCS*. IEEE Computer Society Press, Oct. 2012, pp. 31–40.

[17] N. Bitansky, D. Dachman-Soled, and H. Lin, "Leakage-tolerant computation with input-independent preprocessing," in *CRYPTO 2014, Part II*, ser. LNCS, J. A. Garay and R. Gennaro, Eds., vol. 8617. Springer, Heidelberg, Aug. 2014, pp. 146–163.

[18] D. Dachman-Soled, F.-H. Liu, and H.-S. Zhou, "Leakage-resilient circuits revisited - optimal number of computing components without leak-free hardware," in *EUROCRYPT 2015, Part II*, ser. LNCS, E. Oswald and M. Fischlin, Eds., vol. 9057. Springer, Heidelberg, Apr. 2015, pp. 131–158.

[19] R. Canetti, C. Dwork, M. Naor, and R. Ostrovsky, "Deniable encryption," in *CRYPTO'97*, ser. LNCS, B. S. Kaliski Jr., Ed., vol. 1294. Springer, Heidelberg, Aug. 1997, pp. 90–104.

[20] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters, "Candidate indistinguishability obfuscation and functional encryption for all circuits," in *54th FOCS*. IEEE Computer Society Press, Oct. 2013, pp. 40–49.

[21] A. Sahai and B. Waters, "How to use indistinguishability obfuscation: deniable encryption, and more," in *46th ACM STOC*, D. B. Shmoys, Ed. ACM Press, May / Jun. 2014, pp. 475–484.

[22] R. Canetti, U. Feige, O. Goldreich, and M. Naor, "Adaptively secure multi-party computation," in *28th ACM STOC*. ACM Press, May 1996, pp. 639–648.

[23] S. G. Choi, D. Dachman-Soled, T. Malkin, and H. Wee, "Improved non-committing encryption with applications to adaptively secure protocols," in *ASIACRYPT 2009*, ser. LNCS, M. Matsui, Ed., vol. 5912. Springer, Heidelberg, Dec. 2009, pp. 287–302.

[24] I. Damgård and J. B. Nielsen, "Improved non-committing encryption schemes based on a general complexity assumption," in *CRYPTO 2000*, ser. LNCS, M. Bellare, Ed., vol. 1880. Springer, Heidelberg, Aug. 2000, pp. 432–450.

[25] I. Damgård, Y. Ishai, and M. Kr○igaard, "Perfectly secure multiparty computation and the computational overhead of cryptography," in *EUROCRYPT 2010*, ser. LNCS, H. Gilbert, Ed., vol. 6110. Springer, Heidelberg, May 2010, pp. 445–465.

[26] J. Naor and M. Naor, "Small-bias probability spaces: Efficient constructions and applications," in *22nd ACM STOC*. ACM Press, May 1990, pp. 213–223.

[27] Y. Dodis and A. Smith, "Correcting errors without leaking partial information," in *37th ACM STOC*, H. N. Gabow and R. Fagin, Eds. ACM Press, May 2005, pp. 654–663.

[28] O. Goldreich, *Foundations of Cryptography: Basic Applications*. Cambridge, UK: Cambridge University Press, 2004, vol. 2.

[29] E. Mossel, A. Shpilka, and L. Trevisan, "On e-biased generators in NC0," in *44th FOCS*. IEEE Computer Society Press, Oct. 2003, pp. 136–145.

[30] Y. Ishai, E. Kushilevitz, R. Ostrovsky, and A. Sahai, "Extracting correlations," in *50th FOCS*. IEEE Computer Society Press, Oct. 2009, pp. 261–270.

[31] F. Davì, S. Dziembowski, and D. Venturi, "Leakage-resilient storage," in *Security and Cryptography for Networks, 7th International Conference, SCN 2010, Amalfi, Italy, September 13-15, 2010. Proceedings*, 2010, pp. 121–137. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-15317-4_9

[32] Y. Dodis and A. Smith, "Entropic security and the encryption of high entropy messages," in *TCC 2005*, ser. LNCS, J. Kilian, Ed., vol. 3378. Springer, Heidelberg, Feb. 2005, pp. 556–577.

[33] B. Applebaum, Y. Ishai, and E. Kushilevitz, "On pseudorandom generators with linear stretch in NC^0," *Computational Complexity*, vol. 17, no. 1, pp. 38–69, 2008. [Online]. Available: http://dx.doi.org/10.1007/s00037-007-0237-6

[34] S. Fehr and C. Schaffner, "Randomness extraction via δ-biased masking in the presence of a quantum attacker," in *TCC 2008*, ser. LNCS, R. Canetti, Ed., vol. 4948. Springer, Heidelberg, Mar. 2008, pp. 465–481.

[35] A. Beimel, A. Gabizon, Y. Ishai, E. Kushilevitz, S. Meldgaard, and A. Paskin-Cherniavsky, "Non-interactive secure multiparty computation," in *CRYPTO 2014, Part II*, ser. LNCS, J. A. Garay and R. Gennaro, Eds., vol. 8617. Springer, Heidelberg, Aug. 2014, pp. 387–404.

[36] A. Bogdanov, Y. Ishai, E. Viola, and C. Williamson, "Bounded indistinguishability and the complexity of recovering secrets," *Electronic Colloquium on Computational Complexity (ECCC)*, vol. 22, p. 182, 2015.

[37] N. Alon and Y. Roichman, "Random cayley graphs and expanders," *Random Struct. Algorithms*, vol. 5, no. 2, pp. 271–285, 1994. [Online]. Available: http://dx.doi.org/10.1002/rsa.3240050203

[38] O. Goldreich and A. Wigderson, "Tiny families of functions with random properties: A quality-size trade-off for hashing," *Random Struct. Algorithms*, vol. 11, no. 4, pp. 315–343, 1997. [Online]. Available: http://dx.doi.org/10.1002/(SICI)1098-2418(199712)11:4⟨315::AID-RSA3⟩3.0.CO;2-1

[39] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *SIAM J. Comput.*, vol. 38, no. 1, pp. 97–139, 2008. [Online]. Available: http://dx.doi.org/10.1137/060651380

[40] V. Goyal, Y. Ishai, H. K. Maji, A. Sahai, and A. Sherstov, "Bounded-communication leakage resilience via parity-resilient circuits," Cryptology ePrint Archive, 2016.