On the (Im)possibility of Cryptography with Imperfect Randomness

Yevgeniy Dodis*
New York University

Shien Jin Ong †
Harvard University

Manoj Prabhakaran‡
Princeton University and UCLA

Amit Sahai§
Princeton University and UCLA

Abstract

We investigate the feasibility of a variety of cryptographic tasks with imperfect randomness. The kind of imperfect randomness we consider are entropy sources*, such as those considered by Santha and Vazirani, Chor and Goldreich, and Zuckerman. We show the following:*

- *Certain cryptographic tasks like bit commitment, encryption, secret sharing, zero-knowledge, non-interactive zero-knowledge, and secure two-party computation for any non-trivial function are* **impossible** *to realize if parties have access to entropy sources with slightly less-than-perfect entropy, i.e., sources with imperfect randomness. These results are unconditional and do not rely on any unproven assumption.*
- *On the other hand, based on stronger variants of standard assumptions, secure signature schemes are possible with imperfect entropy sources. As another positive result, we show (without any unproven assumption) that interactive proofs can be made sound with respect to imperfect entropy sources.*

1. Introduction

Randomness is an important concept in computer science. Not surprisingly, a large body of work in theoretical computer science has investigated the requirements on "randomness" in its various roles. In this work, we ask the following fundamental question with regard to cryptography: if "randomness" is to be used in cryptographic protocols, what properties must it have? Indeed, traditional cryptographic protocols are assumed to have access to *perfect random sources*, *i.e.*, sources that output unbiased and independent random bits. However, it is not clear if such perfect randomness is crucial for provable security. We initiate a study on the types of imperfectness in a random source that are tolerable for cryptographic applications.

Is Entropy Sufficient for Randomness? We examine a very natural intuition, which states that all we need for cryptographic protocols is a source of randomness with a guarantee of high *entropy*. In other words, this intuition implies that protocols can be made secure with *any* random source which has a high rate of entropy. We show that this intuition is *false* for many basic and important cryptographic objectives, even when only considering security against computationally efficient adversaries. These results stand in sharp contrast with the fact that entropy is enough for simulating probabilistic polynomial-time algorithms [46, 40, 12, 48]. Intuitively, the reason for this difference comes from the fact that randomized algorithms utilize randomness only for the purposes of *efficiency*, and can in principle be always derandomized (potentially incurring up to exponential penalty in the running time). On the other hand, cryptographic applications fundamentally require randomness to guarantee *security*, and usually cannot even be *defined* without randomness (e.g., if the attacker has no uncertainly about the secrets).

Theory and Practice of Using Randomness. Besides its significant theoretical interest, one motivation for this study concerns the way randomness is generated and used in practice. In practice, it is unrealistic to assume access to perfect random sources. Further it is unrealistic to assume even a specific distribution for the random source. Instead one may have only certain guarantees on the random source (like high entropy rate).

However, it is known how to obtain nearly uniform ran-

* Email: dodis [at] cs [dot] nyu [dot] edu. Research supported in part by NSF CAREER and TC grants.

† Email: shienjin [at] eecs [dot] harvard [dot] edu. Research done while the author was at Princeton University. Research supported in part by the Princeton University Gordon Wu Graduate Fellowship.

‡ Email: mp [at] cs [dot] princeton [dot] edu. Research done while the author was at Princeton University.

§ Email: sahai [at] cs [dot] princeton [dot] edu. Research done while the author was at Princeton University. Research supported in part by NSF ITR and the Alfred P. Sloan Foundation.

dom bits if multiple *independent* sources with high rates of entropy are available [40, 45, 44, 12, 15, 14, 3]. Nevertheless, the assumption of independence between random sources is questionable, especially if the sources are available to a single party locally.

Furthermore, it is not current practice to use multiple independent random sources to build a single perfect random source. Instead, the widely held intuition is that high entropy is sufficient. Indeed, in the applied cryptography community various techniques have been developed for accessing "good" physical sources of randomness, with the focus overwhelmingly on ensuring high *entropy*.[1] This is not surprising considering that the intuitive notion of randomness is almost synonymous with the quantifiable notion of entropy. Thus it is important to understand the theoretical basis, if any, of this intuition, in the context of cryptography.

Imperfect Randomness. Originating from the pioneering work of von Neumann [47], a large amount of research has been devoted towards understanding the applicability of imperfect random sources to the many algorithms and protocols designed to work with perfect randomness. The most straightforward approach to dealing with an imperfect random source is to *deterministically* (and efficiently) extract nearly-perfect randomness from it. Indeed, such results were obtained, in varying extents, for several classes of imperfect random sources. They include various simple "streaming" sources [47, 19, 9, 31], different flavors of "bit-fixing" sources [13, 8, 1, 11, 17, 28], efficiently samplable sources [43], and multiple independent imperfect random sources [40, 45, 44, 12, 15, 14, 3]. While these results are interesting and non-trivial, the above "deterministically extractable" sources assume a lot of structure or independence in the way they generate randomness.

Entropy Sources. A much less restrictive, and arguably more realistic, assumption on the random source would be to assume only that the source contains *some* entropy. We call such sources *entropy sources*. Entropy sources were first introduced by Santha and Vazirani [40], and later generalized by Chor and Goldreich [12], and Zuckerman [48].

In entropy sources with *min-entropy* k, the only guarantee is that any particular string x of length n appears with probability at most 2^{-k} in a sample from that source. Such a source is called an (n, k)-entropy source. For most protocols, where there are multiple parties involved, different parties require multiple samples from a given source. Building on the model of Chor and Goldreich [12], we assume that the source outputs a sequence of blocks $(X_1, X_2, \ldots)$, where each block X_i is an (n, k)-source even conditioned on *any* realization of the all other blocks.[2] We call this source an (n, k)-*entropy block source*.

Note that in the context of entropy sources, a central parameter of interest is the block length n, which specifies "how often" new entropy is guaranteed to be produced by the source. The strongest (most nearly perfect) guarantee would be that *every* bit produced by the source contains new entropy (*i.e.* the block length n is 1). These sources are called *Santha-Vazirani (SV) sources* [40]. We stress that all our impossibility results hold even for SV sources.

The works of [40, 12, 38] show that deterministic randomness extraction of even a single bit is *not* possible from any non-trivial entropy source, including SV sources.

The Thesis of Our Work. In this paper we investigate whether in the context of cryptography, even against computationally efficient adversaries, the notion of randomness is captured by entropy alone. We present the following observations.

(1) A key concept in modern cryptography is *indistinguishability*. In the standard setting with perfect randomness, we know that indistinguishability with computationally unlimited adversaries is only achievable in certain limited settings. However, if we restrict ourselves to computationally efficient adversaries, and make some computational assumptions (like the existence of one-way functions), a new world opens up, allowing for bit commitment, multiple-message encryption, public-key encryption, secret sharing, computational zero-knowledge for all of $\mathbf{NP}$, two-party secure computation, and many other non-trivial protocols.

We consider entropy sources that are only slightly imperfect, such as, in particular, an SV source where each bit's probability of being 0 or 1 is between $1/2 - 1/\mathsf{poly}(\kappa)$ and $1/2 + 1/\mathsf{poly}(\kappa)$, where κ is a security parameter, and $\mathsf{poly}(\kappa)$ denotes an arbitrarily large polynomial. Such a source could have statistical distance which is within any inverse polynomial factor from the uniform distribution. Even for such apparently nearly perfect random sources, we establish our main Lemma: that computationally indistinguishable distributions must be almost identical.

Based on this result, we show that essentially all cryptographic tasks involving some kind of privacy (or "secrecy") *cannot be realized* with respect to entropy sources (including SV sources), *regardless of any computational assumptions* that one is willing to make. In particular, we rule out bit commitment, encryption,

1 For examples, see the following page maintained by D. Wagner: `http://www.cs.berkeley.edu/~daw/rnd/`.

2 The original definitions of [12, 40] are less stringent in that each block is an (n, k)-source conditioned only on the realization of *previous* blocks. We use a stronger formulation also considered recently by [38].

secret sharing, zero-knowledge, non-interactive zero-knowledge, and secure two-party computation for any non-trivial function. In many cases, these (unconditional) impossibility results remain even if some parties receive *independent* or *perfect* sources of randomness, because mutually distrusting parties cannot make use of this independence.

We conclude, surprisingly, that *entropy is not enough* for a useful theory of cryptography.

(2) On the other hand, some applications in cryptography do not rely on the notion of indistinguishability, but only on *unforgeability*. This is the case for authentication tasks. We show that digital signature schemes that are existentially unforgeable against adaptive chosen-message attack, the "gold standard" of authentication, are achievable even with imperfect entropy sources. The assumption we make is the existence of one-way permutations that are secure even when their input comes from a similar entropy source.

(3) Finally, we consider non-cryptographic applications of randomness in protocols, particularly for achieving *soundness* in interactive proof systems. Here we illustrate how to convert any interactive proof system that works for perfect randomness into one that works with very weak block entropy sources. Our transformation is unconditional, round preserving, and results in a public-coin protocol. In particular, it shows that classes like $\mathbf{IP}$ and $\mathbf{AM}$ can be simulated with entropy sources, much like $\mathbf{BPP}$.

Our Results in More Detail. Lemma 3.1, our main technical lemma, forms the basis for all our impossibility results in this paper. In that lemma, we show that if two functions F and G produce computationally indistinguishable outputs when feed any slightly imperfect entropy source as input, then in fact $F(x) = G(x)$ for almost all inputs x. This remains true even if one of the functions gets perfect randomness in addition to the entropy source.

Based on this lemma, we obtain the following impossibility results, which hold for nearly perfect entropy sources such as SV sources with $1/\mathsf{poly}(\kappa)$ bias, where κ is a security parameter, and (trivially therefore), block sources with $n - 1/\mathsf{poly}(\kappa)$ bits of entropy per n-bit block. We stress that *no unproven assumptions* are made in establishing these impossibility results.

- (**Commitment and Encryption.**) First, we rule out bit commitment, even if the receiving party has independent perfect randomness. Another immediate corollary of the main lemma is the impossibility of encryption protocols, symmetric or public key. Here we must assume that both parties share (different blocks of) an entropy source, since if given independent sources, the parties could use variants of two-source extractors [15] to do a secure encryption.
- (**Secret Sharing.**) We rule out secret sharing schemes by showing that even the most basic requirement of such a scheme is unattainable. That is, using only imperfect randomness, it is impossible to distribute a secret to $\ell > 1$ parties in such a way that each party individually will learn nothing about the secret, but all of them combined will be able to retrieve the secret.
- (**Zero-Knowledge.**) A somewhat more sophisticated use of the main lemma allows us to conclude that zero-knowledge proofs (and arguments) only exist for languages in $\mathbf{BPP}$. Our result only requires the prover to have imperfect randomness; the verifier can make use of independent perfect randomness.
- (**Non-Interactive Zero-Knowledge (NIZK).**) For the case of NIZK proof system with respect to a common reference string (CRS), we show that as long as the CRS arises from an entropy source, *even if both prover and verifier have access to independent perfect randomness*, NIZK proofs (and arguments) exist only for $\mathbf{BPP}$ languages.
- (**Secure Two-Party Computation.**) Finally, we show that secure two-party computation is impossible for any *non-trivial* function (non-trivial functions were defined and considered by [4]). This is true even if the two parties hold independent entropy sources. This rules out, in particular, functions such as Oblivious Transfer and the AND operation on two bits.

We also have the following **positive** results:

- (**Digital Signatures.**) We show that digital signature schemes *are achievable* with respect to imperfect entropy sources. The assumption we make is the existence of one-way permutations that are hard to invert when their inputs come from entropy sources. This non-standard assumption is somewhat necessary since the existence of the above mentioned signature schemes would imply the existence of one-way functions that are hard to invert when their inputs come from entropy sources.

  While all standard one-way permutations remain secure against entropy sources with $n - O(\log n)$ bits of entropy, for lower entropy sources a standard one-way permutation could possibly be trivially invertible. We conjecture, nevertheless, that one-way permutations secure against much lower entropy sources exist. Based on this conjecture, and using entropy block sources, we show how to construct a digital signature scheme that is existentially unforgeable against adaptive chosen-message attack. Our construction is an adaption of the construction of Naor and Yung [36].

- (**Interactive Proofs.**) Finally, we also examine interactive proofs with respect to entropy sources. We give a transformation which converts any $\ell(\kappa)$-round interactive proof, where $\ell(\kappa) \leq \mathsf{poly}(\kappa)$ and κ is the input length, which is sound and complete when the verifier has perfect randomness, into an $\ell(\kappa)$-round interactive proof which is sound and complete even when the verifier has any entropy block source with $1/\mathsf{poly}(\kappa)$ entropy per block. Our transformation is a relatively straightforward application of *strong randomness extractors* [37, 32].

Previous Work. The most relevant work to our setting is that of McInnes and Pinkas [35]. They proved that in the setting of computationally unlimited adversaries, one cannot have secure symmetric encryption if the shared key comes from an entropy block source (including SV sources). Our result regarding symmetric encryption could be viewed as a non-trivial extension of their result to efficient adversaries. On the other hand, the result of Dodis and Spencer [18] showed that if the entropy source can have more structure, then some imperfect random sources are sufficient for (one-time) symmetric bit encryption but not for deterministic bit extraction. Additionally, Koshiba [29, 30] considered security definitions for public-key encryption when the encryption algorithm is using an imperfect source (but key generation remains perfect), and showed that in this setting semantic security and indistinguishability are no longer equivalent in general. Our results show that if the key generation is imperfect as well, no security notion for public-key encryption is achievable *at all*.

The question of message authentication in the computationally unbounded setting was explored in [34, 18], who roughly showed that one-time message authentication is possible provided that the entropy rate of the source is greater than $1/2$. In contrast, our signature result constructs a much more complex "multi-time" primitive, for arbitrary entropy rate, but under a strong computational assumption (which is essentially required). Additionally, questions of authentication with respect to imperfect randomness were also considered in the interactive setting by [39], and in a biometric setting by [16]. However, both these works [39, 16] assume that the parties have local access to ideal randomness (but share an imperfectly generated secret key).

Finally, our technique for simulating interactive protocols with weak sources is related to the question of randomness-efficient error/round-reduction in interactive protocols considered by [5, 6, 49].

Future Work. Our results present many fascinating challenges to the theoretical cryptography community. If entropy is not sufficient for cryptography, then what is? Besides independence or structure, are there other characterization of randomness that would allow for computational indistinguishability? Otherwise, is there a "tight" relationship between independence (or structure), and computational indistinguishability? If entropy is really all we can assume, can we obtain weaker levels of security for tasks like encryption, commitment, or secure computation?

2. Preliminaries

For a distribution X over the set $\{0,1\}^n$, we define the *min-entropy* of X to be

$$\mathbf{H}_\infty(X) = \min_{x \in \{0,1\}^n} \left\{ -\log_2(\Pr[X = x]) \right\}.$$

We denote the uniform distribution over $\{0,1\}^n$ as $\mathcal{U}_n$, or simply $\mathcal{U}$ when the domain is clear. To denote the distribution of a random variable X, we write $\{X\}$. The notation $\{F(X,Y)\}$ is shorthand for $\{F(x,y)\}_{x \leftarrow X, y \leftarrow Y}$.

A block source is composed of blocks of equal number of bits. For a block source $X = (\mathcal{X}_1, \ldots, \mathcal{X}_t)$, X_i denotes the i-th block and $\overline{X_i}$ denotes all blocks but the i-th block. The definition of an (n,k)-*block source* that we use was considered in [38], and is a more stringent definition as compared to the one considered by [12]. In our definition, we require that each n-bit block to have min-entropy at least k, even conditioned on any realization of the *all* the other blocks (instead of just the previous blocks, as considered in [12]).

Definition 2.1 ((n,k)-block source). A distribution $X = (\mathcal{X}_1, \ldots, \mathcal{X}_t)$ over $\{0,1\}^{nt}$ is an (n,k)-block source if for all $i = 1, \ldots, t$, and for each $z \in \{0,1\}^{nt-n}$, we have that $\mathbf{H}_\infty(X_i | \overline{X_i} = z) \geq k$.

A *Santha-Vazirani source*, denoted as $\mathrm{SV}(\alpha)$, is special case of an (n,k)-block source with $n = 1$ and $k = -\log_2(1-\alpha)$, where $\alpha \in [0, 1/2]$. (once again, the original definition of [40] only conditioned on prior blocks.)

We use κ to denote the security parameter of our protocols, $\mathsf{poly}(\kappa)$ to represent *any* polynomial, and $\mathrm{neg}(\kappa)$ to denote a negligible function (i.e., $\mathrm{neg}(\kappa) = o(1/\mathsf{poly}(\kappa))$).

For a pair of distributions X and Y, we write $\{X\} \simeq_\varepsilon \{Y\}$ if for every polynomial-sized (in κ) circuit C, we have that $\left|\Pr_{z \leftarrow X}[C(z) = 1] - \Pr_{z \leftarrow Y}[C(z) = 1]\right| \leq \varepsilon(\kappa)$, for all sufficiently large κ. If $\varepsilon(\kappa)$ is a negligible function, then we say that distributions X and Y are *computationally indistinguishable, i.e.,* $\{X\} \cong_{\mathrm{c}} \{Y\}$. For our impossibility results we will use very simple one-bit tests C which simply output the i-th bit of their given input. In other words, if $X \simeq_\varepsilon Y$, then in particular $\left|\Pr_{z \leftarrow X}[z_i = 0] - \Pr_{z \leftarrow Y}[z_i = 0]\right| \leq \varepsilon$, where z_i is the i-th bit of z.

3. Main Lemma

In this section, we prove the main lemma used to establish impossibility results for cryptographic protocols with

imperfect randomness. Informally, this main lemma states that if two functions F and G always produce computationally indistinguishable distributions when fed any (slightly) imperfect entropy source, then the functions must be almost (pointwise) identical. The result still holds if the one of the functions, say F, is probabilistic.

We stress that the main lemma and all our impossibility results (in Section 4) apply to SV sources where each bit is biased away from uniform by only $1/\mathsf{poly}(\kappa)$, where $\mathsf{poly}(\kappa)$ can be an arbitrarily large polynomial.

Lemma 3.1 (Main Lemma). *Let κ be the security parameter, p be any polynomial, and n be any positive integer (including 1). Let Γ be the class of all $(n, n - 1/p(\kappa))$-block sources with t blocks, and $N = nt$. The values of n, t, N' and m are all upperbounded by a polynomial in κ.*

Suppose functions $F\colon \{0,1\}^N \times \{0,1\}^{N'} \to \{0,1\}^m$ and $G\colon \{0,1\}^N \to \{0,1\}^m$ are such that for every distribution $X \in \Gamma$, and for some (arbitrary) distribution Y over $\{0,1\}^{N'}$, we have that $\{F(X,Y)\} \cong_c \{G(X)\}$.[3] Then $\Pr_{(x,y)\leftarrow(\mathcal{U}_N,Y)}[F(x,y) \neq G(x)] = \mathrm{neg}(\kappa)$. Specifically, $\mathrm{neg}(\kappa)$ is at most $O(p(\kappa)^2 m\varepsilon)$, where ε is the best distinguishing advantage between the above distributions.

By setting $N' = 0$, we get the following corollary.

Corollary 3.2. *Let κ, p, Γ, n, t, N and m be as above. Suppose functions $F\colon \{0,1\}^N \to \{0,1\}^m$ and $G\colon \{0,1\}^N \to \{0,1\}^m$ are such that $\{F(X)\} \cong_c \{G(X)\}$ for every $X \in \Gamma$. Then $\Pr_{x\leftarrow\mathcal{U}_N}[F(x) \neq G(x)] = \mathrm{neg}(\kappa)$.*

By considering $(1, 1-1/\mathsf{poly}(\kappa))$-block sources (setting $n = 1$), all our impossibility results in Section 4 extend to Santha-Vazirani [40] sources too.

Corollary 3.3. *Let κ, p, N, N', m, F and G be as in Lemma 3.1. Suppose for every SV $(1/2 - 1/p(\kappa))$ distribution X over $\{0,1\}^N$ and some (arbitrary) distribution Y over $\{0,1\}^{N'}$, we have $\{F(X,Y)\} \cong_c \{G(X)\}$.[3] Then, $\Pr_{(x,y)\leftarrow(\mathcal{U}_N,Y)}[F(x,y) \neq G(x)] = \mathrm{neg}(\kappa)$.*

3.1. Proof of Lemma 3.1 (Main Lemma)

In proving Lemma 3.1, we use an important notion that we call *δ-biased halfspace sources*, which was implicitly defined in the work of Reingold, Vadhan and Wigderson [38].

Definition 3.4. (δ-biased halfspace sources) For $S \subset \{0,1\}^N$ of size $|S| = 2^{N-1}$, and $0 \leq \delta \leq 1/2$, the distribution D_S^δ over $\{0,1\}^N$ is defined as follows: for all $x \in S$, $\Pr_{D_S^\delta}[x] = (1/2 + \delta)2^{-(N-1)}$, and for all $x \notin S$, $\Pr_{D_S^\delta}[x] = (1/2 - \delta)2^{-(N-1)}$.

The collection of all δ-biased halfspace sources is denoted as $D^\delta \stackrel{\text{def}}{=} \{D_S^\delta : S \subset \{0,1\}^N, |S| = 2^{N-1}\}$. First, we prove an analogue of Lemma 3.1 for δ-biased halfspace sources (instead of block sources).

Lemma 3.5. *Let $F\colon \{0,1\}^N \times \{0,1\}^{N'} \to \{0,1\}^m$ and $G\colon \{0,1\}^N \to \{0,1\}^m$. Let Y be some (arbitrary) distribution over $\{0,1\}^{N'}$. Suppose for all δ-biased halfspace sources $X \in D^\delta$ we have that $\{F(X,Y)\} \simeq_\varepsilon \{G(X)\}$. Then $\Pr_{(x,y)\leftarrow(\mathcal{U}_N,Y)}[F(x,y) \neq G(x)] \leq m\varepsilon\delta^{-2}$.*

Proof sketch. Fix a position $i \in [1,m]$, and let $f(x,y) \stackrel{\text{def}}{=} F_i(x,y)$, the i-th bit of $F(x,y)$. Similarly let $g(x) \stackrel{\text{def}}{=} G_i(x)$.

We would like to bound the probability that $f(x,y) \neq g(x)$ when $x \leftarrow \mathcal{U}_N$ and $y \leftarrow Y$. Define the probabilities p_{00}, p_{01}, p_{10} and p_{11} as

$$p_{bb'} = \Pr_{(x,y)\leftarrow(\mathcal{U}_N,Y)}[f(x,y) = b \wedge g(x) = b'].$$

We can assume without loss of generality that $\Pr_{x\leftarrow\mathcal{U}_N}[g(x) = 0] \leq 1/2$. That is $p_{00} + p_{10} \leq 1/2$. The quantity we want to bound is $p_{10} + p_{01}$.

It can be shown that there exists a set S, with $|S| = 2^{N-1}$ and $\{x : g(x) = 0\} \subseteq S \subseteq \{x : g(x) = 0\} \cup \{x : \Pr_{(x',y)\leftarrow(\mathcal{U}_N,Y)}[f(x',y) = 0 | x' = x] \leq \varepsilon/(2\delta)\}$. Applying the hypothesis of the lemma to the distribution $X = D_S^\delta$, and we have that

$$\Pr[g(D_S^\delta) = 0] = (1 + 2\delta)(p_{00} + p_{10}),$$
$$\Pr[f(D_S^\delta, Y) = 0] \leq (1 + 2\delta)(p_{00} + \tau) + (1 - 2\delta)p_{01}.$$

By the hypothesis of the lemma, the above two probabilities differ by at most ε. From this and the fact that $p_{01} \leq p_{10} + \varepsilon$ (obtained by observing that when the hypothesis of the lemma holds for all $X \in D^\delta$, it will hold for $X = \mathcal{U}_N$ too), it can be shown that $p_{10} + p_{01} \leq \varepsilon\delta^{-2}$.

Thus for any $i \in [m]$, $\Pr[F_i(\mathcal{U}_N, Y) \neq G_i(\mathcal{U}_N)] = p_{10} + p_{01} \leq \varepsilon\delta^{-2}$. Our lemma follows by using a union bound over all $i \in [m]$. □

The next lemma shows that δ-biased halfspace sources are in fact very strong entropy sources.

Lemma 3.6 ([38]). *For any positive integer n, the distribution D_S^δ is an $(n, n - \log_2((1+2\delta)/(1-2\delta)))$-block source with $t = N/n$ blocks.*

To complete the proof of Lemma 3.1 (Main Lemma), we set $\delta = 1/(8p(\kappa) + 2)$. Then, one can check that $\log_2((1+2\delta)/(1-2\delta)) \leq 1/p(\kappa)$. Hence for all set S, the distribution D_S^δ is an $(n, n - 1/p(\kappa))$-block source with t blocks. By Lemma 3.5, we get $\Pr[F(\mathcal{U}_N, Y) \neq G(\mathcal{U}_N)] \leq m\varepsilon\delta^{-2} = \mathrm{neg}(\kappa)$, since $\varepsilon = \mathrm{neg}(\kappa)$, and m and $1/\delta$ are bounded by a polynomial in κ.

3 For simplicity, the reader may assume that Y is independent of the first input to F, *i.e.*, the joint distribution (X, Y) is a product distribution. But in fact, Y can be *dependent* on the first input in the following manner: for each $x \in \{0,1\}^N$, Y specifies the distribution on $\{0,1\}^{N'}$ conditioned on the first input being x.

4. Impossibility of Certain Cryptographic Protocols with Imperfect Randomness

For this section, let κ denote the desired security parameter of the protocols, and let n be any positive integer denoting the block length of the block source. We show that even with slightly imperfect randomness, *i.e.,* $(n, n - 1/\mathsf{poly}(\kappa))$-block sources, fundamental cryptographic protocols like commitment, encryption, zero-knowledge proofs, non-interactive zero-knowledge proofs, and two-party secure computation are *not realizable*, no matter what computational cryptographic assumptions we are willing to make.

We stress that all our impossibility results hold for $\mathrm{SV}(1/2 - 1/\mathsf{poly}(\kappa))$ sources, simply by setting $n = 1$.

4.1. Commitment and Encryption

Theorem 4.1 (Impossibility of commitment). *Suppose the sender's (committing party) only random source is an $(n, n - 1/\mathsf{poly}(\kappa))$-block source. Then commitment with (security parameter κ) is impossible.*

Note that the impossibility of commitment as stated in Theorem 4.1 holds even if the receiving party is given access to uniform randomness.

Proof sketch. Let Y be any $(n, n - 1/\mathsf{poly}(\kappa))$-block source. Suppose the sender commits to a bit b by sending $\mathrm{Commit}(0; r)$ where $r \leftarrow Y$. The hiding property of the commitment requires that a commitment to 0 and a commitment to 1 be computationally indistinguishable, namely $\{\mathrm{Commit}(0; r)\}_{r \leftarrow Y} \cong_c \{\mathrm{Commit}(1; r)\}_{r \leftarrow Y}$. By Corollary 3.2 of the Main Lemma, both functions $\mathrm{Commit}(0; \cdot)$ and $\mathrm{Commit}(1; \cdot)$ must be almost identical. In other words, for almost all r, $\mathrm{Commit}(0; r) = \mathrm{Commit}(1; r)$. This violates the (computational) binding property of the commitment since the sender can trivially decommit to both bits 0 and 1. The proof extends to interactive commitment protocols by considering transcripts instead of commitments, and to the case when the receiver has an independent source of uniform randomness, by considering all non-uniform receivers which work with all possible fixed random-tapes (then for each such receiver the transcript functions must be almost identical). □

Theorem 4.2 (Impossibility of encryption). *Suppose both parties are given a* single *source Y as the only source of randomness (prior to and during message transmission). Then, there* do not exist *semantically secure encryption protocols (with security parameter κ) that are secure for every $(n, n - 1/\mathsf{poly}(\kappa))$-block source Y.*

The proof of Theorem 4.2 is similar to the proof of Theorem 4.1, and hence omitted.

4.2. Secret Sharing

Secret sharing schemes are used in cryptographic applications to distribute a secret to ℓ parties in such a way that only if k of them collude would they manage to obtain the secret. Even if $k - 1$ of them collude, they should not gain any computational advantage in guessing the secret. If a secret sharing scheme satisfies that requirement, we say that is has a (k, ℓ)-threshold. A formal definition of such a scheme is given in [21].

With perfect uniform randomness, Shamir [41] presented a (k, ℓ)-threshold scheme for any $k \in [2, \ell]$. However if we only have imperfect randomness, we prove that it is *impossible* to distribute a secret to ℓ parties in such a way that each party individually will learn nothing about the secret, but all of them combined will be able to retrieve the secret.

Theorem 4.3. *For any $2 \leq k \leq \ell$, there does not exist a (k, ℓ)-threshold secret sharing scheme (with security parameter κ) that uses only randomness from a $(n, n - 1/\mathsf{poly}(\kappa))$-block source.*

Proof sketch. Because the secret sharing algorithm has access to only imperfect randomness, by Corollary 3.2, it must be the case that all the shares of secret s will be identical to all the shares of some other secret s' (with high probability). But since $s \neq s'$, it will be impossible to reconstruct the secret even if all ℓ parties collude. □

4.3. Zero-Knowledge

Zero-knowledge proofs [25] are interactive proof systems that yield no additional knowledge other than the fact that the statement proven is true. In the uniform randomness setting, it has been shown by a series of works [23, 27, 7] that zero-knowledge proofs exactly characterize $\mathbf{PSPACE}$, the class of problems solvable by polynomial-space bounded machines. On the other hand, with only slightly imperfect randomness, we prove that (auxiliary-input) zero-knowledge proofs are *impossible* for languages not in $\mathbf{BPP}$.

To formalize this notion of zero-knowledge with imperfect randomness, let Y be an (n, k)-block source. The only source of the prover's randomness is a *single sample* of imperfect randomness $y \leftarrow Y$. We allow both the verifier and the simulator to have access to uniform randomness, noting that the impossibility result still holds in this case.

In this model, *giving the prover's random string x to the verifier may potentially leak knowledge*. This is because the verifier does not know what the distribution Y is. The only guarantee on Y is that it is an (n, k)-block source. Hence, the simulator is required to be *universal* with respect to Y. In other words, the simulator needs to output a *single* prover-verifier transcript for all possible (n, k)-block sources Y given as the prover's randomness.

Contrast this to the uniform randomness setting, where giving a uniform random string to the verifier leaks no knowledge. After all, the verifier can obtain the random string by itself (since uniform independent randomness is assumed to be freely available in that setting).

Our main result on the impossibility of zero-knowledge is stated as follows.

Theorem 4.4. *If a language L has an auxiliary-input zero-knowledge proof (with security parameter κ) and the prover's only random source is an imperfect $(n, n - 1/\mathsf{poly}(\kappa))$-block source, then $L \in \mathbf{BPP}$.*

The above impossibility result extends to rule out zero-knowledge arguments.[4] The proof of Theorem 4.4 relies on the following lemma.

Lemma 4.5. *Let $F\colon \{0,1\}^N \to \{0,1\}^m$ and let Γ be the set of all $(n, n - 1/\mathsf{poly}(\kappa))$-block sources of length N. If $\{F(Y_1)\} \cong_c \{F(Y_2)\}$ for every $Y_1, Y_2 \in \Gamma$, then there exists an $\alpha \in \{0,1\}^m$ s.t. $\Pr_{y \leftarrow \mathcal{U}_N}[F(y) = \alpha] > 1 - \mathrm{neg}(\kappa)$.*

Proof sketch. Set $H(\cdot, b) \stackrel{\text{def}}{=} F(b)$, and observe that $\{H(Y_2, Y_1)\} \equiv \{F(Y_1)\} \cong_c \{F(Y_2)\}$. Applying Lemma 3.1 (Main Lemma), we can show that for some fixed β, $H(\cdot, \beta) = F(\cdot)$ almost everywhere. But observe that $\alpha \stackrel{\text{def}}{=} F(\beta) = H(\cdot, \beta)$ takes on a constant value. □

Proof sketch of Theorem 4.4. Let Γ be the set of all $(n, n - 1/\mathsf{poly}(\kappa))$-block sources. Our first step is to show that the prover P must be almost deterministic. Assume that the verifier sends the first message. Consider the cheating verifier V^* which outputs as its first message the auxiliary input z and halts afterwards. We claim that the prover's first message $P_1(x, Y, z)$, is almost deterministic. Let S_{V^*} be the simulator for V^*. Then the zero-knowledge condition (on the first pair of messages) implies that

$$\begin{aligned}\{S_{V^*}(x, z, \mathcal{U})\} &\cong_c \{\langle P(x, Y), V^*(x, z, \mathcal{U})\rangle\} \\ &\equiv \{(z, P_1(x, Y, z))\},\end{aligned}$$

for all distributions $Y \in \Gamma$. Therefore, for any $Y_1, Y_2 \in \Gamma$, we have that $\{(z, P_1(x, Y_1, z))\} \cong_c \{(z, P_1(x, Y_2, z))\}$. By Lemma 4.5, there exists a message π such that $\Pr_{y \leftarrow \mathcal{U}}[P_1(x, y, z) = \pi] > 1 - \mathrm{neg}(\kappa)$. This means that the prover's first message is almost deterministic. Repeating this argument inductively, we find that all the prover's messages must be almost deterministic.

Having shown that the prover is almost deterministic, we can use the techniques of Goldreich and Oren [24] to show that any auxiliary-input zero-knowledge proof system with almost deterministic provers can only decide languages in $\mathbf{BPP}$. □

4 The soundness in an argument is only guaranteed against computationally efficient cheating provers. An impossibility result for arguments is stronger than that of proofs, since any proof system is, by definition, also an argument.

4.4. Non-Interactive Zero-Knowledge

Non-interactive zero-knowledge proof systems (NIZK) were introduced by Blum, Feldman and Micali [10]. The NIZK model allows the prover and the verifier to share a *common random string* (CRS). In the perfect randomness setting, the CRS is a uniform random string chosen by a trusted party. The prover sends a *single* message, and then the verifier will decide to accept or reject based on the prover's message, the CRS, and its own randomness. Feige, Lapidot and Shamir [20] showed that all languages in $\mathbf{NP}$ possess NIZK proofs if one-way permutations exist.

In the imperfect randomness setting, the CRS is chosen by a trusted party from an $(n, n - 1/\mathsf{poly}(\kappa))$-block source. We prove that NIZK is impossible in this setting.

Theorem 4.6. *Let $\langle P, V\rangle$ be an NIZK protocol for a language L. Suppose the CRS is generated from a source Y using a function G, that is CRS $= G(x)$, where $x \leftarrow Y$.[5] If the NIZK protocol (with security parameter κ) is secure for every $(n, n - 1/\mathsf{poly}(\kappa))$-block source Y, then $L \in \mathbf{BPP}$.*

Our impossibility results holds even when the prover and verifier are each allowed to have access to uniform randomness. In addition, it is also possible to rule out NIZK arguments[4] for languages outside $\mathbf{BPP}$, if G is efficiently invertible in the following sense: there is an efficient procedure G^{-1} such that $G(G^{-1}(z)) = z$ for all $z \in \mathrm{Range}(G)$, and $\{G^{-1}(G(x))\}_{x \leftarrow \mathcal{U}} \equiv \mathcal{U}$.

Proof sketch of Theorem 4.6. Let the simulator for the NIZK proof system be $S \stackrel{\text{def}}{=} (F, \Pi)$, where F generates the CRS and Π generates the proof. We claim that the following algorithm A is a $\mathbf{BPP}$ procedure for deciding the language L.

Algorithm A: On input x, select $y \leftarrow \mathcal{U}$ and $r \leftarrow \mathcal{U}$. Set $\rho = F(x, y, r)$ and $\pi = \Pi(x, y, r)$. If $\rho = G(y)$ and $V(x, \rho, \pi) = 1$, then *accept*. Else *reject*.

For $x \notin L$, it can be shown that the cheating prover strategy defined by $P^*(x, \rho) = \Pi(x, G^{-1}(\rho), \mathcal{U})$ will succeed in making V accept with at least the same probability that A accepts x. Hence the soundness condition guarantees that A rejects x with high probability.

For $x \in L$, the zero-knowledge condition stipulates that $\{F(x, Y, \mathcal{U})\} \cong_c \{G(Y)\}$, and hence by Lemma 3.1 (Main Lemma), $F(x, y, r) = G(y)$ for almost all y and r. This means that given $y \leftarrow \mathcal{U}$, the simulator is almost always forced to produce the exact copy of the CRS, which is $G(y)$.[6] And since the "proof" generated by Π is computationally indistinguishable from the honest prover's proof, algorithm A will accept $x \in L$ with high probability. □

5 The function G can be *any* (even uncomputable) function.

6 Contrast this to the uniform randomness setting where the simulator usually manipulates the distribution of the CRS to gain an advantage over a cheating prover.

4.5. Two-Party Secure Computation

Let $f\colon S_1 \times S_2 \to S_3$ be a two-argument finite function, that is all S_1, S_2, and S_3 are finite sets. Let Alice and Bob be the parties involved in computing f. The private input to Alice and Bob are x_A and x_B respectively. They wish to securely compute the value of $f(x_A, x_B)$, in a way that will not allow the other party to gain knowledge of their private inputs. We consider an asymmetric notion of secure computation whereby only Bob needs to output $f(x_A, x_B)$.[7]

Informally, we say that an interactive protocol between Alice and Bob securely computes $f(x_A, x_B)$ if after the interaction, the following two conditions hold.

1. Bob learns the right value of $f(x_A, x_B)$ but no matter how he tries to cheat, he will learn nothing about x_A which is not already implied by x_B and $f(x_A, x_B)$.
2. Alice learns nothing about x_B no matter how she tries to cheat.

We refer the reader to [21] for the formal definition of two-party secure computation.

A function f is said to be *trivial* if there exists a two-party secure computation protocol such that both honest parties are deterministic, and remains secure even if the malicious party is computationally unbounded. Beimel, Malkin and Micali [4] gave a *deterministic one-round* protocol computing any trivial function f. The protocol just involves Alice sending a single message to Bob. In addition, they gave an exact combinatorial characterization of trivial functions.

Theorem 4.7 ([4]). *A function $f\colon S_1 \times S_2 \to S_3$ is* trivial *iff there* do not *exist $a_0, a_1 \in S_1$ and $b_0, b_1 \in S_2$, such that $f(a_0, b_0) = f(a_1, b_0)$ and $f(a_0, b_1) \neq f(a_1, b_1)$.*

In the uniform randomness model, Goldreich, Micali and Wigderson [22] proved that all functions are securely computable if trapdoor permutations exist. With only imperfect randomness, we show that the only trivial functions are securely computable.

Theorem 4.8. *Assume the two parties are given* independent *$(n, n - 1/\mathsf{poly}(\kappa))$-block sources. If there exists two-party secure computation protocols (with security parameter κ) computing a two-argument finite function f in the malicious model, then f is trivial.*

While the above theorem rules out secure computation in the malicious setting, we cannot do much better even in the *honest-but-curious* model, in which the security guarantee is only for honest execution of the protocol.

7 In the malicious model, it is unreasonable to expect both parties to always be able to output the correct evaluation of the function, because the first party that obtains the output of the function can abort.

Theorem 4.9 (impossibility in honest-but-curious setting). *Let Y and Z be random sources of Alice and Bob respectively, and f be a two-argument finite function. If there exists two-party secure computation (with security parameter κ) of f in the honest-but-curious model that works for all $(n, n - 1/\mathsf{poly}(\kappa))$-block source $Y \circ Z$, then f is trivial.*

Our result is tight, in the sense that if we assume independence of Y and Z, we can use extractors to obtain independent private uniform randomness for both parties [12, 15, 14]. And with private uniform randomness, all functions are securely computable [22]. Therefore, if the two parties are given *independent* (n, k)-block sources, for $k > n/2 + \omega(\log n)$, then all functions are securely computable in the honest-but-curious model (if trapdoor permutations exists).

5. Secure Signature Schemes with Imperfect Random Sources

Turning to our positive results, we construct signature schemes that remain secure even if our random source is only guaranteed to be a (n, k)-block source for $k < n$. The only cryptographic assumption we make is the existence of a one-way permutation (OWP) that remains secure with imperfect random sources. Our signature scheme will be *existentially unforgeable under chosen message attack*. The verification of our signature scheme is deterministic, but the signer will be probabilistic and stateful.

Informally we say a protocol or a function is (n, k)-secure if it remains secure even using an imperfect random source that is only guaranteed to be a (n, k)-block source. In particular, a one-way permutation $f\colon \{0,1\}^{O(n)} \to \{0,1\}^{O(n)}$ is (n, k)-secure if there does not exists an efficient algorithm capable of inverting f even when the input to f is sampled from any (n, k)-block source. The following is our main theorem regarding signature schemes.

Theorem 5.1. *If (n, k)-secure one-way permutations exist, then (n, k)-secure signature schemes exist.*

The construction of our signature scheme is very similar to that of Naor and Yung [36]. Our main observation is that we are able to do a reduction (in the imperfect random sources model) from an adversary A breaking the signature scheme to another related adversary A' breaking the OWP.

Necessity of the non-standard assumption. While we use a stronger variant of OWP to construct signature schemes with imperfect randomness, we note that (n, k)-secure signature schemes readily imply the existence of (n, k)-secure one-way functions. This is because the key generation algorithm can be viewed as a one-way function, with the input being the randomness used to generate the public/secret

keys pair and the output being the public key. This fact suggests that the non-standard assumption of (n,k)-secure OWP is needed as a basis for the construction of (n,k)-secure signature schemes.

We note that $(n, n-O(\log n))$-secure OWP are equivalent to standard OWP. Furthermore, $(n, n-n^{\varepsilon})$-secure OWP, for some $\varepsilon > 0$, follow from the recently popular assumption of *strongly intractable* OWP. Strongly intractable OWP are permutations that are hard-to-invert even by $2^{n^{\Omega(1)}}$-sized circuits.

6. Interactive Protocols With Weak Sources

A long line of research on explicit extractor construction has shown that the class of probabilistic polynomial-time algorithms (**BPP**) can be simulated using (n,k)-block sources, as long as n is bounded by a polynomial in the input length and $k \geq n^{\Omega(1)}$ (*e.g.,* see [32]). In this section, we show that the same conclusion holds for a much richer class of interactive protocols.

Interactive Protocols with Uniform Randomness. In the standard interactive proof protocol [2, 25], a computationally unbounded prover P needs to convince a probabilistic polynomial-time verifier V (with access to uniform randomness) membership in the language L. That is, for $x \in L$, we have $\Pr[\langle P,V\rangle(x) = 1] \geq 2/3$ *(completeness)*. And for $x \notin L$, and for any cheating prover P^*, we have $\Pr[\langle P^*,V\rangle(x) = 1] \leq 1/3$ *(soundness)*. Let the class $\mathbf{IP}[t]$ denote languages possessing a t-round (private-coin) interactive proof protocol. If all the verifier's messages consists of just random coin tosses, we call such an interactive protocol *public-coin*, and denote the corresponding class by $\mathbf{AM}[t]$, the class of t-round public-coin interactive proof protocol. We know that $\mathbf{IP}[t] = \mathbf{AM}[t]$ ([26]), that $\mathbf{IP}[\mathsf{constant}] = \mathbf{AM}[2]$ ([2]), and that $\mathbf{AM}[\mathsf{poly}] = \mathbf{IP}[\mathsf{poly}] = \mathbf{PSPACE}$ ([33, 42]).

Interactive Protocols with Imperfect Randomness. Analogous to the case of probabilistic algorithms with imperfect randomness, we consider interactive protocols where the verifier V have access to only imperfect randomness with the only guarantee of being an (n,k)-block source. We denote the corresponding classes by $\mathbf{IP}_{weak}[t]$ and $\mathbf{AM}_{weak}[t]$. While these definitions technically should depend on n and k, but we will show momentarily that as long as $n \leq \mathsf{poly}(|x|)$ and $k \geq 1/\mathsf{poly}(|x|)$, where x is the common input to the interactive protocol, this will not make any difference. Specifically, we show the following.

Theorem 6.1. *For any t, $\mathbf{AM}_{weak}[t] = \mathbf{AM}[t] = \mathbf{IP}[t] = \mathbf{IP}_{weak}[t]$. Thus, $\mathbf{IP}[\mathsf{constant}] = \mathbf{AM}_{weak}[2]$ and $\mathbf{AM}_{weak}[\mathsf{poly}] = \mathbf{IP}_{weak}[\mathsf{poly}] = \mathbf{PSPACE}$.*

Proof sketch. Notice, it suffices to show $\mathbf{AM}[t] \subseteq \mathbf{AM}_{weak}[t]$, as then $\mathbf{IP}_{weak}[t] \subseteq \mathbf{IP}[t] = \mathbf{AM}[t] \subseteq \mathbf{AM}_{weak}[t] \subseteq \mathbf{IP}_{weak}[t]$. We only sketch our transformation below, leaving the proof to the full version.

Take any $L \in \mathbf{AM}[t]$ which has a t-round $\mathbf{AM}$-protocol with completeness $11/12$ and soundness $1/12$, where the verifier V send c uniform random bits per round. We will use the notion of *strong randomness extractors* [37] to make a new protocol between new prover P' and new verifier V'. Specifically, for any error ε and min-entropy $m > c + O(\log(1/\varepsilon))$, there exists [32] an efficient strong extractor $\mathsf{Ext} : \{0,1\}^N \times \{0,1\}^d \to \{0,1\}^c$ with seed length $d = O(\log N + \log(1/\varepsilon))$, such that given any (N,m)-source X, the output of $\mathsf{Ext}(X, \mathcal{U}_d)$ is ε-close to $\mathcal{U}_c$, even if conditioned on the seed value. We set $\varepsilon = 1/12t, m = c + O(\log(1/\varepsilon)), N = n\lceil m/k\rceil$ and $d = O(\log N + \log(1/\varepsilon))$, and view our (n,k)-block source as an (N,m)-block source (by grouping together $\lceil m/k\rceil$ original blocks). Notice, the values $N, 2^d, m, t$ are all polynomial in the input length $|x|$.

Now, given our (N,m)-block source $X = (X_1 \ldots X_t)$, we let $R_i^s = \mathsf{Ext}(X_i, s)$ be the value extracted from X_i on seed s. In round i, our new $\mathbf{AM}_{weak}$-verifier V' will send his block X_i, while the prover P' will respond with $2^d = \mathsf{poly}(|x|)$ responses A_i^s which the original prover P would send on the V's challenges $R_1^s \ldots R_i^s$. At the end, V' computes the fraction Z (w.r.t. s) of accepting computations (according to V) and accepts if $Z > 1/2$. □

ACKNOWLEDGMENTS. The authors thank Omer Reingold and Salil Vadhan for sharing their result (joint with Avi Wigderson) on the non-extractability of Santha-Vazirani sources, which allowed us to considerably simplify the proof of our Main Lemma for SV sources. The authors also thank Oded Goldreich, Shafi Goldwasser, Neeraj Kayal, Omer Reingold and Salil Vadhan for helpful comments.

References

[1] M. Ajtai and N. Linial. The influence of large coalitions. *Combinatorica*, 13(2):129–145, 1993.

[2] L. Babai and S. Moran. Arthur-Merlin games: A randomized proof system and a hierarchy of complexity classes. *J. Comput. Syst. Sci.*, 36(2):254–276, 1988.

[3] B. Barak, R. Impagliazzo, and A. Wigderson. Extracting randomness from few independent sources. In *Proc. 45th FOCS*, 2004.

[4] A. Beimel, T. Malkin, and S. Micali. The all-or-nothing nature of two-party secure computation. In *Proc. CRYPTO '99*, pages 80–97, 1999.

[5] M. Bellare, O. Goldreich, and S. Goldwasser. Randomness in interactive proofs. *Comput. Complex.*, 3(4):319–354, 1993.

[6] M. Bellare and J. Rompel. Randomness-effi cient oblivious sampling. In *Proc. 35th FOCS*, pages 276–287, 1994.

[7] M. Ben-Or, O. Goldreich, S. Goldwasser, J. Håstad, J. Kilian, S. Micali, and P. Rogaway. Everything provable is provable in zero-knowledge. In *Proc. CRYPTO '88*, pages 37–56, 1988.

[8] C. H. Bennett, G. Brassard, and J.-M. Robert. Privacy amplification by public discussion. *SIAM J. Comput.*, 17(2):210–229, 1988.

[9] M. Blum. Independent unbiased coin flips from a correlated biased source—a finite state Markov chain. *Combinatorica*, 6(2):97–108, 1986.

[10] M. Blum, P. Feldman, and S. Micali. Non-interactive zero-knowledge and its applications. In *Proc. 20th STOC*, pages 103–112, 1988.

[11] R. Canetti, Y. Dodis, S. Halevi, E. Kushilevitz, and A. Sahai. Exposure-resilient functions and all-or-nothing transforms. In *Proc. EUROCRYPT '00*, pages 453–469, 2000.

[12] B. Chor and O. Goldreich. Unbiased bits from sources of weak randomness and probabilistic communication complexity. *SIAM J. Comput.*, 17(2):230–261, 1988.

[13] B. Chor, O. Goldreich, J. Hastad, J. Friedman, S. Rudich, and R. Smolensky. The bit extraction problem of t-resilient functions. In *Proc. 26th FOCS*, pages 396–407. IEEE, 1985.

[14] Y. Dodis, A. Elbaz, R. Oliveira, and R. Raz. Improved randomness extraction from two independent sources. In *Proc. RANDOM '04*, 2004.

[15] Y. Dodis and R. Oliveira. On extracting private randomness over a public channel. In *Proc. RANDOM '03*, pages 252–263, 2003.

[16] Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In *Proc. EUROCRYPT '04*, pages 523–540, 2004.

[17] Y. Dodis, A. Sahai, and A. Smith. On perfect and adaptive security in exposure-resilient cryptography. In *Proc. EUROCRYPT '01*, pages 301–324, 2001.

[18] Y. Dodis and J. Spencer. On the (non)universality of the one-time pad. In *Proc. 43rd FOCS*, pages 376–388, 2002.

[19] P. Elias. The efficient construction of an unbiased random sequence. *Ann. Math. Stat.*, 43(2):865–870, 1972.

[20] U. Feige, D. Lapidot, and A. Shamir. Multiple non-interactive zero knowledge proofs under general assumptions. *SIAM J. Comput.*, 29(1):1–28, 1999.

[21] O. Goldreich. *Foundations of cryptography*, volume 2. Cambridge University Press, Cambridge, 2004. Basic applications.

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

[23] O. Goldreich, S. Micali, and A. Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *J. ACM*, 38(1):691–729, 1991.

[24] O. Goldreich and Y. Oren. Definitions and properties of zero-knowledge proof systems. *J. Cryptology*, 7(1):1–32, 1994.

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

[26] S. Goldwasser and M. Sipser. Private coins versus public coins in interactive proof systems. *Advances in Computing Research*, 5:73–90, 1989.

[27] R. Impagliazzo and M. Yung. Direct minimum-knowledge computations. In *Proc. CRYPTO '87*, pages 40–51, 1987.

[28] J. Kamp and D. Zuckerman. Deterministic extractors for bit-fixing sources and exposure-resilient cryptography. In *Proc. 35th FOCS*, pages 92–101, 2003.

[29] T. Koshiba. A new aspect for security notions: Secure randomness in public-key encryption schemes. In *Proc. 4th PKC*, pages 87–103, 2001.

[30] T. Koshiba. On sufficient randomness for secure public-key cryptosystems. In *Proc. 5th PKC*, pages 34–47, 2002.

[31] D. Lichtenstein, N. Linial, and M. Saks. Some extremal problems arising from discrete control processes. *Combinatorica*, 9(3):269–287, 1989.

[32] C.-J. Lu, O. Reingold, S. Vadhan, and A. Wigderson. Extractors: optimal up to constant factors. In *Proc. 35th STOC*, pages 602–611, 2003.

[33] C. Lund, L. Fortnow, H. Karloff, and N. Nisan. Algebraic methods for interactive proof systems. *J. ACM*, 39(4):859–868, 1992.

[34] U. Maurer and S. Wolf. Privacy amplification secure against active adversaries. In *Proc. CRYPTO '97*, pages 307–321, 1997.

[35] J. L. McInnes and B. Pinkas. On the impossibility of private key cryptography with weakly random keys. In *Proc. CRYPTO '90*, pages 421–436, 1991.

[36] M. Naor and M. Yung. Universal one-way hash functions and their cryptographic applications. In *Proc. 21th STOC*, pages 33–43, 1988.

[37] N. Nisan and D. Zuckerman. Randomness is linear in space. *J. Comput. Syst. Sci.*, 52(1):43–52, 1996.

[38] O. Reingold, S. Vadhan, and A. Wigderson. A note on extracting randomness from Santha-Vazirani sources. Unpublished manuscript, 2004.

[39] R. Renner and S. Wolf. Unconditional authenticity and privacy from an arbitrarily weak secret. In *Proc. CRYPTO '03*, pages 78–95, 2003.

[40] M. Santha and U. V. Vazirani. Generating quasi-random sequences from semi-random sources. *J. Comput. Syst. Sci.*, 33(1):75–87, 1986.

[41] A. Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, 1979.

[42] A. Shamir. IP = PSPACE. *J. ACM*, 39(4):869–877, 1992.

[43] L. Trevisan and S. Vadhan. Extracting randomness from samplable distributions. In *Proc. 41st FOCS*, pages 32–42, 2000.

[44] U. V. Vazirani. Efficiency considerations in using semi-random sources. In *Proc. 19th STOC*, pages 160–168, 1987.

[45] U. V. Vazirani. Strong communication complexity or generating quasirandom sequences from two communicating semirandom sources. *Combinatorica*, 7(4):375–392, 1987.

[46] U. V. Vazirani and V. V. Vazirani. Random polynomial time is equal to slightly-random polynomial time. In *Proc. 26th FOCS*, pages 417–428, 1985.

[47] J. von Neumann. Various techniques used in connection with random digits. *National Bureau of Standards, Applied Mathematics Series*, 12:36–38, 1951.

[48] D. Zuckerman. Simulating BPP using a general weak random source. *Algorithmica*, 16(4/5):367–391, 1996.

[49] D. Zuckerman. Randomness-optimal oblivious sampling. *Random. Struct. Algor.*, 11(4):345–367, 1997.