Black-Box Reductions and Separations in Cryptography

Marc Fischlin

Darmstadt University of Technology, Germany
www.cryptoplexity.de
marc.fischlin@cryptoplexity.de

Abstract. Cryptographic constructions of one primitive or protocol from another one usually come with a reductionist security proof, in the sense that the reduction turns any adversary breaking the derived scheme into a successful adversary against the underlying scheme. Very often the reduction is black-box in the sense that it only looks at the input/output behavior of the adversary and of the underlying primitive. Here we survey the power and the limitations of such black-box reductions, and take a closer look at the recent method of meta-reductions.

1 Introduction

Since the beginning of modern cryptography in the 70's the design methodology for cryptographic protocols has shifted from ad-hoc constructions and "security by obscurity" techniques to well-founded approaches. This transition shows in the agreed-upon methodology to provide clean attack models and security goals of a protocol, and to give a rigorous proof that the protocol meets these goals. Here, the term "proof" should be understood from a reductionist viewpoint, saying that any successful adversary breaking a cryptographic scheme would entail the efficient break of a presumably hard primitive.

Today a special type of proof, called *black-box reduction*, is pervasive in cryptography and provides a very powerful tool to analyze protocols. Roughly, a reduction is black-box if it does not use any internals of the adversary beyond the input and output behavior, and analogously if nothing about the structure of the underlying primitive except for its basic properties is exploited (such reductions are called fully black-box [RTV04]). It turns out that a vast number of cryptographic primitives such as one-way functions, pseudorandom generators [HILL99], and pseudorandom functions [GGM86] can all be derived from each other in a black-box way. Starting with a result by Impagliazzo and Rudich [IR89], though, for some important problems it has been proven that black-box reductions cannot exist. These negative results are summarized under the name *black-box separations*.

In this paper we survey the three main techniques for black-box separation results, namely, the relativization technique [IR89], the two-oracle technique [HR04], and the increasingly more popular meta-reduction technique [BV98]. We start with an overview about black-box constructions and, after having reviewed the three separation techniques, we also briefly discuss *non*-black-box constructions to indicate potential limitations and bypasses of black-box separation results.

2 Black-Box Constructions

In this section we look at the positive cases of constructions which are black-box and the equivalence class of symmetric-key primitives, called Minicrypt [Imp95].

2.1 One-Way Functions are Necessary

Most of today's cryptography is impossible without assuming the existence of (cryptographic) one-way functions. Of course, we can symmetrically encrypt messages securely with the One-Time Pad encryption, but as shown by Shannon [Sha49] this basically requires the key to be of equal length as the message. If, on the other hand, one tries to securely encrypt messages which are larger than the key, then this immediately implies the existence of one-way functions, as formally shown by Impagliazzo and Luby [IL89]. In this paper, Impagliazzo and Luby also show further primitives to imply one-way functions, like bit commitments, (private-key) identification, and coin-flipping over phone.

It should be mentioned that all these implications are constructive in the sense that one can build a concrete one-way function f given the primitive in question, even given the primitive as a black-box only. For instance, for a semantically-secure symmetric encryption scheme Enc which allows to encrypt messages of twice the length as the key, the one-way function is given by $f(k,m) = \mathsf{Enc}(k,m)||m$. Furthermore, the reduction from the one-wayness to the security of the underlying primitive treats both the adversary and the primitive as black-boxes, such that the overall constructions are also called fully black-box [RTV04].

The implications also mean that most cryptographic primitives are not known to exist for sure. That is, the existence of (cryptographic) one-way functions implies (worst-case) one-way functions and thus $\mathcal{P} \neq \mathcal{NP}$. In other words, $\mathcal{P} \neq \mathcal{NP}$ is necessary for numerous cryptographic tasks. It is, however, currently not known if it is also sufficient [AGGM06, BT06].

2.2 One-Way Functions are Sufficient for Minicrypt

In a sense, one-way functions appear to be "very low" in the hierarchy of assumptions. They are not only necessary for most cryptographic tasks, but they also suffice to build a lot of cryptographic primitives. In a series of papers it has been shown that one-way functions imply pseudorandom generators [HILL99], that such pseudorandom generators imply pseudorandom functions [GGM86], and that pseudorandom functions imply pseudorandom permutations [LR88]. Once one has the powerful pseudorandom functions then other primitives like message authentication codes (MACs), private-key encryption, and private-key identification are derived easily. All these constructions and reductions are of the fully black-box type.

Impagliazzo [Imp95] calls the world in which we have cryptographic one-way functions, but no public-key cryptography, "Minicrypt"; as opposed to "Cryptomania" in which we have all the power of public-key encryption. In Minicrypt, we can still do a remarkably number of cryptographic tasks like sending messages securely to parties which we have met before; only secure communication with strangers in impossible then. Somewhat unexpected, another very interesting primitive which can also be built from one-way functions and thus lies in Minicrypt, are secure digital signature schemes. This has been shown in a sequence of papers [NY89, Rom90], again in the fully black-box sense. The noteworthy property here is that, structurally, digital signatures are of course related to public-key primitives; existentially, though, they belong to the family of symmetric-key primitives.

3 Black-Box Separations

In this section we review the three main techniques for black-box separations ans the questions which primitives lie (presumably) outside of Minicrypt.

3.1 Relativizing Reductions: Separating Key Agreement from One-Way Functions

In their seminal paper, Impagliazzo and Rudich [IR89] show that one cannot base (even weakly) secure key agreement on one-way functions. More precisely, they first use a (random) permutation oracle to implement a one-way permutation. This oracle can later be derandomized and one "good" oracle can be found by standard counting arguments and the Borel-Cantelli lemma (see [IR89] for details). In the next step they show that relative to an $\mathcal{NP}$-oracle no key agreement protocol based on the random permutation oracle can be secure. (A simplified version of this fact for the case of perfectly complete key agreement can be found in [BKSY11].) Put differently, there cannot exist *relativizing* constructions of key agreement from one-way permutations, i.e., where the security of the construction remains intact in the presence of an arbitrary oracle.

As pointed out by [IR89, Sim98, RTV04] relativing reductions where the relativizing oracle allows for an embedding of an $\mathcal{NP}$-oracle —or more generally, any $\mathcal{PSPACE}$-oracle, such that any "standard" cryptography besides the one-way permutation can be broken— can be shown to rule out so-called $\forall\exists$ semi-black-box reductions [RTV04]. Roughly, these are efficient reductions which turn efficient successful adversaries for one scheme into an adversary for the other one, where both the adversary and the reduction have oracle access to the primitive oracle, potentially also containing the embedded $\mathcal{NP}$ or $\mathcal{PSPACE}$ oracle. Since such reductions only use the underlying primitive as a black-box, but can depend in the adversary's implementation, separations on this level are "somewhat less black-box" than in the case of fully black-box reductions, strengthening the separation result.

Relativizing separations can be found in [IR89, Sim98, GKM+00, Fis02, Hof11]. In particular, Rudich [Rud92] used this technique to separate k-round key agreement from any $(k+1)$-round key agreement, implying an infinite hierarchy of primitive classes.

3.2 Fully Black-Box Reductions: The Two-Oracle Technique by Hsiao and Reyzin

Since relativizing reductions (with embedding) are equivalent to $\forall\exists$ semi-black-box reductions [RTV04] showing impossibility results is much more challenging than for the fully black-box case. Hence, Hsiao and Reyzin [HR04] introduced the idea of moving from relativizing reductions to fully black-box reductions, and use a so-called two-oracle technique. The idea is roughly to have an oracle Ω which is used to implement the primitive Q we would like to have, say, a one-way function or permutation. The second oracle Π is used to break the primitive P which we are trying to build out of the one given through Ω. For a separation it then suffices to show that one can implement Q from Ω (ignoring Π), such that that for all algorithms $\mathcal{R}$ there exists some adversary $\mathcal{A}$ such that $\mathcal{A}^{\Pi}$ breaks P, but $\mathcal{R}^{\mathcal{A}^{\Pi,\Omega},\Omega}$ cannot break Q. Note that in the latter case $\mathcal{R}$ only has access to Π through the black-box access to $\mathcal{A}$, although most proofs later use a universal $\mathcal{A}$ which basically merely runs Π, such that this essentially boils down to show that $\mathcal{R}^{\Pi,\Omega}$ should not be able to break Q.

Because the two-oracle technique allows for easier separations it became quite popular and

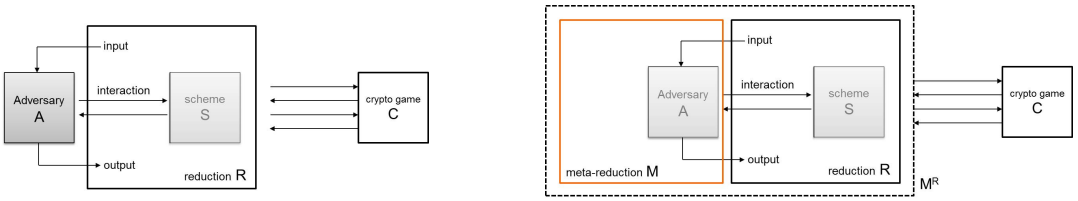

Figure 1: (a) shows the reduction $\mathcal{R}$ turning a successful adversary $\mathcal{A}$ against scheme $\mathcal{S}$ into a successful attacker $\mathcal{R}^{\mathcal{A}}$ against a cryptographic game C, by simulating the scheme $\mathcal{S}$; (b) shows the meta-reduction $\mathcal{M}$ simulating the adversary $\mathcal{A}$ and turning $\mathcal{R}$ into a successful algorithm $\mathcal{M}^{\mathcal{R}}$ against C directly.

has been applied more often in recent papers. Examples include [HR04, DOP05, BCFW09, FLR+10, FS12].

3.3 The Meta-Reduction Technique

Recently, a new kind of black-box separation technique has gained significant attention, called *meta-reductions* [BV98].[1] Roughly, a meta-reduction is a "reduction against the reduction". The situation is depicted in Figure 1: The reduction $\mathcal{R}$ is given black-box access to an adversary $\mathcal{A}$, which supposedly attacks a scheme $\mathcal{S}$, but where $\mathcal{S}$ is now simulated by the reduction. The reduction itself is supposed to break a so-called cryptographic game C with the help of $\mathcal{A}$. This game usually models any falsifiable assumption [Nao03], including assumptions like computing discrete logarithms or inverting the RSA function. We note that, in order to avoid trivial reductions like to the security of the scheme itself, the game C often consists of less rounds than the interactive phase of the scheme.

The meta-reduction now simulates the adversarial part in order to turn $\mathcal{R}$ in a black-box manner into an efficient and successful algorithm $\mathcal{M}^{\mathcal{R}}$ against C directly, without reference to an allegedly successful adversary $\mathcal{A}$. Note that this clearly requires the existence of a successful adversary $\mathcal{A}$ against $\mathcal{S}$ in the first place, or else the reduction $\mathcal{R}$ would not need to break C at all. Usually, one can build such an (inefficient) adversary by using exhaustive search for the secrets, and then one needs to make sure that the efficient $\mathcal{M}$ can still replace $\mathcal{A}$. Similarly to the case of zero-knowledge, where the efficient simulator can mimic the behavior of the all-powerful prover, the meta-reduction's advantage over the adversary here is that it can rewind the reduction (or potentially take advantage of its code or behavior). Overall, if the meta-reduction is sufficiently close to $\mathcal{A}$ from $\mathcal{R}$'s perspective, it follows that the probability for $\mathcal{M}^{\mathcal{R}}$ breaking C is close to the one of $\mathcal{R}^{\mathcal{A}}$.

The advantage of meta-reductions over the other separation types is that this technique usually only makes black-box use of the adversary, but works with arbitrary primitives. The technique therefore applies to cases where one, say, seeks to show that certain constructions cannot be based on the RSA assumption. As such, this separation technique is "below" fully black-box reductions and dual to $(\forall\exists)$semi-black-box reductions. On the other hand, it seems that the method is mainly suitable for interactive protocols in which the scheme can be queried first, before the adversary is required to produce an output. Examples include unforgeability of signature schemes under chosen-message attacks or chosen-ciphertext security for encryption schemes.

[1] Albeit the idea appears in [BV98] it seems as if the term meta-reduction has only been mentioned later in [Bro05] and [PV05].

In summary, the meta-reduction technique usually consist of the following three steps:

1. Design an all-powerful adversary $\mathcal{A}$ which breaks the scheme.

   For example, in the signature case let $\mathcal{A}$ first compute a secret key sk^* from pk, then let it query the signature oracle to collect signatures (note that this step is only necessary to build the meta-reduction), and finally let $\mathcal{A}$ compute a forgery.

2. Replace the (inefficient) adversary by the efficient meta-reduction.

   This is usually done by carefully rewinding the reduction at appropriate places in the query phase. To prevent the reduction from making further queries the rewinding is usually done when the reduction does not make queries to the game C. This may also require further conditions on the reduction to prevent the nested-rewinding problem (the reduction seeking to reset the adversary while the meta-reduction aims to reset the reduction). This problem may yield an exponential blow-up and is known from the area of zero-knowledge [DNS04].

3. Show that the meta-reduction's behavior is sufficiently close to the one of the all-powerful adversary.

   This step is usually the most challenging step as the meta-reduction's output is somewhat closer entangled with the reduction's state than the adversary's behavior, due to the rewinding.

With these steps it follows that $\mathcal{M}^{\mathcal{R}}$ breaks the game C with probability close to the reduction $\mathcal{R}^{\mathcal{A}}$ (given adversary $\mathcal{A}$).

Meta-reductions have been successfully applied in a number of cases since [BV98], such as [Cor02, Bro05, PV05, FS10, Pas11, GW11, DHT12, Seu12]. It is clear that the exact use of meta-reductions differ, e.g., some results also impose restrictions on the primitives and work for black-box groups only.

4 Non-Black-Box Constructions

In this section we mention some non-black-box constructions resp. reductions. Both examples stem form the area of zero-knowledge proofs but the issue is in principle not restricted to this area.

4.1 Karp Reductions are Non-Black-Box

The first examples touches the issue of Karp reductions between problems. Recall that a Karp reduction from one language A to another language B is a deterministic polynomial-time algorithm k such that $x \in A \iff \mathsf{k}(x) \in B$. If such an algorithm exist then we write $A \leq_p B$, intuitively meaning that the problem B is at least as hard as A (in the sense that any decision algorithm for B would immediately yield a decider for A). Cook and Levin have shown that the satisfiability is complete for $\mathcal{NP}$, i.e., any other problem $A \in \mathcal{NP}$ reduces to the satisfiability problem. This reduction, however, makes use of the (Turing machine) code of the algorithm M_A deciding A by representing its computation state as a boolean formula. In other words, the Karp reduction of A to the satisfiability problem requires access to the code for deciding A.

The code-dependence is exactly where the black-box property for cryptographic purposes may break down. Given an arbitrary one-way function f and, say, proving in zero-knowledge

that one knows a pre-image to some y under f, one would reduce this problem to some $\mathcal{NP}$-complete language L for which such a proof is known via a Karp reduction, and to run the zero-knowledge protocol for L.[2] However, the reduction from f to L would then require knowledge of the code of f and does not apply to black-box constructions for f. Note that it may still be possible to find direct zero-knowledge proofs for specific one-way functions, like the Schnorr proof for discrete logarithms [Sch91], or find other alternatives to the Karp reduction to L. We finally note that Brakerski et al. [BKSY11] recently introduced special zero-knowledge oracles to argue about separations in the presence of such proofs.

4.2 Barak's Non-Black-Box Zero-Knowledge Proofs

The second example is based on a non-black-box use of the adversary. Barak [Bar01] designs a zero-knowledge proof based on *non-black-box* use of the adversary which overcomes previous black-box impossibility results. Neglecting many technical subtleties, the protocol to prove $x \in L$ is roughly as follows. The protocol first runs an initialization phase whose only purpose is to give the zero-knowledge simulator some freedom. In this phase, the prover commits to the all-zero string π and the verifier send a random string r. Now the prover and the verifier engage in a witness-indistinguishable protocol [FS90] that $x \in L$ or that the commitment π describes a program that predicts the verifier's string r.

A malicious prover cannot take advantage of the initialization phase —predicting the uknown random string r remains infeasible— and thus really needs to prove $x \in L$ in the second step. A zero-knowledge simulator against a malicious verifier, on the other hand, can simply use the non-black-box access to the verifier's code and its randomness, and commit to the verifier's program (with fixed randomness) on behalf of the prover. By the hiding property of the commitment scheme this is indistinguishable from a commitment to zeros. It is clear that this code π predicts r correctly, such that the simulator can use π as the witness in the second part of the proof to faithfully simulate these steps, even without knowing a witness to $x \in L$ or by using the usual rewinding techniques. The zero-knowledge property follows from the hiding of the commitment and the witness indistinguishability of the second part.

5 Conclusion

Black-box separations (of any kind) are today thought of as good indications that one cannot derive one primitive out of the other. But they can also been viewed as a shortcoming of the proof technique itself. A few non-black-box constructions do exist, and one option to circumvent black-box separations may be to use more non-black-box techniques. For example, Harnik and Naor [HN06] showed that, using a complexity-theoretic assumption, one can build (in a non-black-box way) collision-resistant hash functions out of one-way functions, allowing to bypass Simon's black-box separation result for this case [Sim98]. Unfortunately, Fortnow and Santhanam [FS08] later showed that the assumption is unlikely to hold, or else the polynomial hierarchy collapses. Still, it remains open to explore the limitations of black-box separations via non-black-box techniques, or to strengthen the separation results along the line of Brakerski et al. [BKSY11].

[2] Speaking of zero-knowledge proofs of *knowledge* in our example, one would need to ensure that the Karp reduction is such that a witness extracted from the proof for L also allows to recover a pre-image for f; this is usually the case and such reductions are sometimes called Levin reductions.

Acknowledgments

I would like to thank Paul Baecher and Christina Brzuska for discussions about black-box reductions, and the Africacrypt program committee and its chair, Serge Vaudenay, for inviting me to present the topic.

References

[AGGM06] Adi Akavia, Oded Goldreich, Shafi Goldwasser, and Dana Moshkovitz. On basing one-way functions on NP-hardness. In Jon M. Kleinberg, editor, *38th ACM STOC*, pages 701–710, Seattle, Washington, USA, May 21–23, 2006. ACM Press. (Cited on page 2.)

[Bar01] Boaz Barak. How to go beyond the black-box simulation barrier. In *42nd FOCS*, pages 106–115, Las Vegas, Nevada, USA, October 14–17, 2001. IEEE Computer Society Press. (Cited on page 6.)

[BCFW09] Alexandra Boldyreva, David Cash, Marc Fischlin, and Bogdan Warinschi. Foundations of non-malleable hash and one-way functions. In Mitsuru Matsui, editor, *ASIACRYPT 2009*, volume 5912 of *LNCS*, pages 524–541, Tokyo, Japan, December 6–10, 2009. Springer, Berlin, Germany. (Cited on page 4.)

[BKSY11] Zvika Brakerski, Jonathan Katz, Gil Segev, and Arkady Yerukhimovich. Limits on the power of zero-knowledge proofs in cryptographic constructions. In Yuval Ishai, editor, *TCC 2011*, volume 6597 of *LNCS*, pages 559–578, Providence, RI, USA, March 28–30, 2011. Springer, Berlin, Germany. (Cited on pages 3 and 6.)

[Bro05] Daniel R. L. Brown. Breaking rsa may be as difficult as factoring. *IACR Cryptology ePrint Archive*, 2005. (Cited on pages 4 and 5.)

[BT06] Andrej Bogdanov and Luca Trevisan. On worst-case to average-case reductions for np problems. *SIAM J. Comput.*, 36(4):1119–1159, 2006. (Cited on page 2.)

[BV98] Dan Boneh and Ramarathnam Venkatesan. Breaking RSA may not be equivalent to factoring. In Kaisa Nyberg, editor, *EUROCRYPT'98*, volume 1403 of *LNCS*, pages 59–71, Espoo, Finland, May 31 – June 4, 1998. Springer, Berlin, Germany. (Cited on pages 1, 4, and 5.)

[Cor02] Jean-Sébastien Coron. Security proof for partial-domain hash signature schemes. In Moti Yung, editor, *CRYPTO 2002*, volume 2442 of *LNCS*, pages 613–626, Santa Barbara, CA, USA, August 18–22, 2002. Springer, Berlin, Germany. (Cited on page 5.)

[DHT12] Yevgeniy Dodis, Iftach Haitner, and Aris Tentes. On the instantiability of hash-and-sign rsa signatures. In *TCC*, volume 7194 of *Lecture Notes in Computer Science*, pages 112–132. Springer, 2012. (Cited on page 5.)

[DNS04] Cynthia Dwork, Moni Naor, and Amit Sahai. Concurrent zero-knowledge. *J. ACM*, 51(6):851–898, 2004. (Cited on page 5.)

[DOP05] Yevgeniy Dodis, Roberto Oliveira, and Krzysztof Pietrzak. On the generic insecurity of the full domain hash. In Victor Shoup, editor, *CRYPTO 2005*, volume 3621 of *LNCS*, pages 449–466, Santa Barbara, CA, USA, August 14–18, 2005. Springer, Berlin, Germany. (Cited on page 4.)

[Fis02] Marc Fischlin. On the impossibility of constructing non-interactive statistically-secret protocols from any trapdoor one-way function. In Bart Preneel, editor, *CT-RSA 2002*, volume 2271 of *LNCS*, pages 79–95, San Jose, CA, USA, February 18–22, 2002. Springer, Berlin, Germany. (Cited on page 3.)

[FLR+10] Marc Fischlin, Anja Lehmann, Thomas Ristenpart, Thomas Shrimpton, Martijn Stam, and Stefano Tessaro. Random oracles with(out) programmability. In Masayuki Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 303–320, Singapore, December 5–9, 2010. Springer, Berlin, Germany. (Cited on page 4.)

[FS90] Uriel Feige and Adi Shamir. Witness indistinguishable and witness hiding protocols. In *STOC*, pages 416–426. ACM, 1990. (Cited on page 6.)

[FS08] Lance Fortnow and Rahul Santhanam. Infeasibility of instance compression and succinct PCPs for NP. In Richard E. Ladner and Cynthia Dwork, editors, *40th ACM STOC*, pages 133–142, Victoria, British Columbia, Canada, May 17–20, 2008. ACM Press. (Cited on page 6.)

[FS10] Marc Fischlin and Dominique Schröder. On the impossibility of three-move blind signature schemes. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 197–215, French Riviera, May 30 – June 3, 2010. Springer, Berlin, Germany. (Cited on page 5.)

[FS12] Dario Fiore and Dominique Schröder. Uniqueness is a different story: Impossibility of verifiable random functions from trapdoor permutations. In *TCC*, volume 7194 of *Lecture Notes in Computer Science*, pages 636–653. Springer, 2012. (Cited on page 4.)

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *Journal of the ACM*, 33:792–807, 1986. (Cited on pages 1 and 2.)

[GKM+00] Yael Gertner, Sampath Kannan, Tal Malkin, Omer Reingold, and Mahesh Viswanathan. The relationship between public key encryption and oblivious transfer. In *41st FOCS*, pages 325–335, Redondo Beach, California, USA, November 12–14, 2000. IEEE Computer Society Press. (Cited on page 3.)

[GW11] Craig Gentry and Daniel Wichs. Separating succinct non-interactive arguments from all falsifiable assumptions. In Lance Fortnow and Salil P. Vadhan, editors, *43rd ACM STOC*, pages 99–108, San Jose, California, USA, June 6–8, 2011. ACM Press. (Cited on page 5.)

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999. (Cited on pages 1 and 2.)

[HN06] Danny Harnik and Moni Naor. On the compressibility of NP instances and cryptographic applications. In *47th FOCS*, pages 719–728, Berkeley, CA, USA, October 21–24, 2006. IEEE Computer Society Press. (Cited on page 6.)

[Hof11] Dennis Hofheinz. Possibility and impossibility results for selective decommitments. *Journal of Cryptology*, 24(3):470–516, July 2011. (Cited on page 3.)

[HR04] Chun-Yuan Hsiao and Leonid Reyzin. Finding collisions on a public road, or do secure hash functions need secret coins? In Matthew Franklin, editor, *CRYPTO 2004*, volume 3152 of *LNCS*, pages 92–105, Santa Barbara, CA, USA, August 15–19, 2004. Springer, Berlin, Germany. (Cited on pages 1, 3, and 4.)

[IL89] Russell Impagliazzo and Michael Luby. One-way functions are essential for complexity-based cryptography. In *30th FOCS*, pages 230–235, Research Triangle Park, North Carolina, October 30 – November 1, 1989. IEEE Computer Society Press. (Cited on page 2.)

[Imp95] Russell Impagliazzo. A personal view of average-case complexity. In *Structure in Complexity Theory Conference*, pages 134–147, 1995. (Cited on page 2.)

[IR89] Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *21st ACM STOC*, pages 44–61, Seattle, Washington, USA, May 15–17, 1989. ACM Press. (Cited on pages 1 and 3.)

[LR88] Michael Luby and Charles Rackoff. How to construct pseudorandom permutations from pseudorandom functions. *SIAM Journal on Computing*, 17(2), 1988. (Cited on page 2.)

[Nao03] Moni Naor. On cryptographic assumptions and challenges (invited talk). In Dan Boneh, editor, *CRYPTO 2003*, volume 2729 of *LNCS*, pages 96–109, Santa Barbara, CA, USA, August 17–21, 2003. Springer, Berlin, Germany. (Cited on page 4.)

[NY89] Moni Naor and Moti Yung. Universal one-way hash functions and their cryptographic applications. In *21st ACM STOC*, pages 33–43, Seattle, Washington, USA, May 15–17, 1989. ACM Press. (Cited on page 2.)

[Pas11] Rafael Pass. Limits of provable security from standard assumptions. In Lance Fortnow and Salil P. Vadhan, editors, *43rd ACM STOC*, pages 109–118, San Jose, California, USA, June 6–8, 2011. ACM Press. (Cited on page 5.)

[PV05] Pascal Paillier and Damien Vergnaud. Discrete-log-based signatures may not be equivalent to discrete log. In Bimal K. Roy, editor, *ASIACRYPT 2005*, volume 3788 of *LNCS*, pages 1–20, Chennai, India, December 4–8, 2005. Springer, Berlin, Germany. (Cited on pages 4 and 5.)

[Rom90] John Rompel. One-way functions are necessary and sufficient for secure signatures. In *22nd ACM STOC*, pages 387–394, Baltimore, Maryland, USA, May 14–16, 1990. ACM Press. (Cited on page 2.)

[RTV04] Omer Reingold, Luca Trevisan, and Salil P. Vadhan. Notions of reducibility between cryptographic primitives. In Moni Naor, editor, *TCC 2004*, volume 2951 of *LNCS*, pages 1–20, Cambridge, MA, USA, February 19–21, 2004. Springer, Berlin, Germany. (Cited on pages 1, 2, and 3.)

[Rud92] Steven Rudich. The use of interaction in public cryptosystems (extended abstract). In Joan Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 242–251, Santa Barbara, CA, USA, August 11–15, 1992. Springer, Berlin, Germany. (Cited on page 3.)

[Sch91] Claus-Peter Schnorr. Efficient signature generation by smart cards. *Journal of Cryptology*, 4(3):161–174, 1991. (Cited on page 6.)

[Seu12] Yannick Seurin. On the exact security of schnorr-type signatures in the random oracle model. In *EUROCRYPT*, volume 7237 of *Lecture Notes in Computer Science*, pages 554–571. Springer, 2012. (Cited on page 5.)

[Sha49] Claude E. Shannon. Communication theory of secrecy systems. *Bell Systems Technical Journal*, 28(4):656–715, 1949. (Cited on page 2.)

[Sim98] Daniel R. Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? In Kaisa Nyberg, editor, *EUROCRYPT'98*, volume 1403 of *LNCS*, pages 334–345, Espoo, Finland, May 31 – June 4, 1998. Springer, Berlin, Germany. (Cited on pages 3 and 6.)