DIMACS Technical Report 2002-22
May 2002

Bounds on the Efficiency of Encryption and Digital Signatures

by

Rosario Gennaro[1] Yael Gertner[2] Jonathan Katz[3]

[1] `rosario@watson.ibm.com`. IBM T.J. Watson Research Center.
[2] `ygertner@saul.cis.upenn.edu`. Department of Computer and Information Science, University of Pennsylvania.
[3] `jkatz@cs.columbia.edu`. Department of Computer Science, Columbia University. A portion of this work was supported by DIMACS.

DIMACS is a partnership of Rutgers University, Princeton University, AT&T Labs-Research, Bell Labs, NEC Research Institute and Telcordia Technologies (formerly Bellcore).

DIMACS was founded as an NSF Science and Technology Center, and also receives support from the New Jersey Commission on Science and Technology.

ABSTRACT

A central focus of modern cryptography is to investigate the weakest possible assumptions under which various cryptographic algorithms exist. Typically, a proof that a "weak" primitive (e.g., a one-way function) implies the existence of some "strong" algorithm (e.g., a private-key encryption scheme) proceeds by giving an explicit construction of the latter from the former. Beyond merely showing such a construction, an equally important research direction is to explore the *efficiency* of the construction. One might argue that this line of research has become even more important now that minimal assumptions are known for many (but not all) algorithms of interest.

Protocols for encryption (in both the public- and private-key setting) and for digital signatures are fundamental to cryptography. In this work, we show the first lower bounds on the efficiency of constructions of these protocols based on black-box access to one-way or trapdoor one-way permutations. If S is the assumed security of the permutation π (i.e., no adversary of size S can "break" π in the appropriate sense on a fraction larger than $1/S$ of its inputs), our results show that:

- Any public-key encryption algorithm for m-bit messages must query π at least $\Omega(m/\log S)$ times. This matches the known upper bound.

- Any private-key encryption algorithm for m-bit messages which uses a k-bit key must query π at least $\Omega(\frac{m-k}{\log S})$ times. This matches the known upper bound.

- Any signature verification algorithm for m-bit messages must query π at least $\Omega(m/\log S)$ times.

We prove our results in an extension of the Impagliazzo-Rudich model. That is, we show that any black-box construction beating our lower bounds would imply the unconditional existence of a one-way function.

1 Introduction

Research in cryptography generally proceeds along two lines: (1) constructing protocols of interest based on the weakest possible computational assumptions and (2) improving the efficiency of currently-known constructions. Although much success has been achieved for each goal separately, it remains challenging to find protocols which are both efficient and also based on general assumptions (such as one-way functions).

As an example, consider public-key encryption. The most efficient construction of a public-key encryption scheme based on trapdoor permutations requires $\mathcal{O}(m/\log k)$ invocations of the permutation to encrypt an m-bit message [11, 10, 24], where k is the security parameter. On the other hand, schemes based on *specific* assumptions can be much more efficient; i.e., the El Gamal encryption scheme [5] (whose security is based on the decisional Diffie-Hellman assumption) requires only a single function application to encrypt $O(k)$ bits. Similar gaps in efficiency exist for constructions of many other algorithms including pseudorandom generators (PRGs), universal one-way hash functions (UOWHFs), private-key encryption schemes, and digital signatures.

Kim, Simon and Tetali [16] began to investigate the reason for this disparity by giving (non-tight) lower bounds on the efficiency of constructions of UOWHFs from one-way permutations. Extending these results, Gennaro and Trevisan [6] have shown that currently-known constructions of UOWHFs based on one-way permutations are in fact *optimal*. They also show a similar result for the case of PRGs. Thus, the efficiency of these algorithms can be improved only by relying on stronger assumptions.

Their work, however, does not address the efficiency of some fundamental algorithms of cryptography: encryption schemes and digital signature algorithms. Indeed, Gennaro and Trevisan left as an open question whether efficiency bounds can be shown for other cryptographic algorithms. In particular, they ask whether known constructions of public-key encryption schemes based on trapdoor permutations are optimal.

We answer their question here in the affirmative by showing a lower bound on the efficiency of any public-key encryption scheme given black-box access to trapdoor permutations. In addition, we show lower bounds for constructions of private-key encryption schemes and digital signature schemes given black-box access to one-way permutations. Our lower bounds match known upper bounds for the case of encryption, showing that currently-known constructions are indeed optimal.

1.1 Our results

Informally, we say that a one-way (or trapdoor) permutation $\pi : \{0,1\}^n \to \{0,1\}^n$ has security S if any circuit A of size smaller than S inverts π with probability less than $1/S$ (for concreteness, one can think of S as a slightly super-polynomial function of n, such as $n^{\log n}$, but our results hold for any choice of S). Given this definition, our results may be stated as follows.

Encryption. A public key encryption (PKE) scheme is said to be semantically-secure [11] if for any two messages M_0, M_1 the distribution over encryptions of M_0 is computationally indistinguishable from the distribution over encryptions of M_1, even when given the public key as input. A similar definition (but with no public key) holds for the case of private-key encryption. More formal definitions appear in Section 2.3.

If m is the length of the message to be encrypted, the best known construction of a public-key encryption scheme based on trapdoor permutations requires $\mathcal{O}(m/\log S)$ invocations of a permutation with security S [11, 10, 24]. In this construction, a message $M = b_1 \cdots b_m$ is encrypted bitwise by choosing random $x_1, \ldots, x_m$ and sending

$$f(x_1), HC_f(x_1) \oplus b_1, \ldots, f(x_m), HC_f(x_m) \oplus b_m,$$

where $HC_f(\cdot)$ represents a hard-core bit [10] for the trapdoor permutation f. Later improvements allow encrypting $\log S$ bits per invocation of f [10]. We call this specific paradigm for encryption the *hard-core bit paradigm*.

A related paradigm may be used for private-key encryption. Assume the sender and receiver share a key sk of length $k = \Omega(\log S)$ and want to encrypt a message of length m. Fix a PRG $G : \{0,1\}^k \rightarrow \{0,1\}^m$. Encryption of message M is done by sending $G(sk) \oplus M$. The known upper bound of $\mathcal{O}(\frac{m-k}{\log S})$ invocations required for G [3, 10] gives an upper bound on the efficiency of this encryption scheme.

We show that *any* black-box construction of a public-key encryption scheme based on trapdoor permutations *must* query the oracle $\Omega(m/\log S)$ times. Indeed, we show that a construction requiring fewer than this many invocations of the permutation to encrypt an m-bit message can be used to explicitly construct a one-way function. Our lower bound applies to the number of invocations made by the encryption algorithm, *in addition to* any invocations made during the key generation phase, and is therefore tight. Similarly, we show that any construction of a private-key encryption scheme based on one-way permutations which shares a key of length k and requires fewer than $\Omega(\frac{m-k}{\log S})$ invocations of the permutation to encrypt an m-bit message can also be used to give an explicit construction of a one-way function.

Signature Schemes. A signature scheme is said to be existentially unforgeable against chosen message attacks [12] if no efficient algorithm can forge a valid signature on a new message after seeing polynomially many signatures on messages of its choice. In this work, we prove lower bounds on the efficiency of the verification algorithm for *one-time signature schemes* (making our results stronger, since any scheme satisfying the strong definition of security [12] may trivially be used as a one-time signature scheme).

If m is the length of the message to be signed, the best known construction of a signature scheme based on one-way permutations requires $\mathcal{O}(m/\log S)$ invocations of a permutation with security S ([2], building on [18]).

We prove that, for *any* construction of a signature scheme, the verification algorithm must query the permutation oracle $\Omega(m/\log S)$ times. As before, we show that any construction beating this bound can be used to give an explicit construction of a one-way function.

1.2 Overview of our Techniques

Our results hold in the oracle separation model of Impagliazzo and Rudich [15] (see [6, 8] for interesting discussions about this model) which is used to separate primitives under black-box reductions. These are an important subclass of reductions, since almost all reductions in cryptography are black box.

Intuitively, a black-box reduction of B to A is a construction of B out of A that ignores the internal structure of the implementation of A. Furthermore, the security of the resulting implementation of B is based solely on the security of the implementation of A. Black-box reductions work relative to any oracle. Therefore, one way to prove that a black-box reduction of B to A is impossible is to prove the existence of an oracle relative to which A exist whereas B does not. Since the initial work of Impagliazzo and Rudich [15], many other separation results [23, 7, 8, 22, 21, 13] in this model have appeared. We use a related technique (used previously [16, 6]) here. To show a lower bound on the efficiency of constructions of B from A, we show an oracle for A relative to which a construction of B beating the bound would yield the unconditional existence of something beyond current techniques; e.g., a one-way function.

A corollary of [6] is that a one-way permutation with security S cannot have more than $O(\log S)$ hard-core bits. This immediately yields a lower bound of $\Omega(m/\log S)$ invocations for PKE encryption schemes *that follow the hard-core bit paradigm* (as in [11, 10, 24]). Yet, we were unable to extend this idea to obtain a lower bound that holds for *any* PKE scheme. Indeed, the proof of our lower bound for the case of encryption follows a different approach. We show that an encryption scheme that makes fewer than the prescribed number of queries can be used to build a private-key encryption scheme in which the key is shorter than the message. Using a previous result of Impagliazzo and Luby [14], this implies the unconditional existence of a one-way function.

Perhaps one reason for this difference between our techniques and those of Gennaro and Trevisan is that our setting can be seen as more complex than that considered previously. Specifically, the primitives analyzed, PRGs and UOWHFs, have only one "party", or algorithm, to consider. Therefore, in that scenario the oracle can be replaced by a random string which is given as input to the algorithm. Public-key encryption and digital signature schemes, however, involve two parties. Here, the only input shared by the parties is the public key which is available to the adversary as well! Therefore, we cannot simply use the technique of replacing the oracle by a random string given to one of the parties without also giving this string to the adversary (in which case the proof of security no longer holds).

An interesting feature of our lower bound for signature schemes is that it bounds the efficiency of the *verification algorithm*. There are several practical applications that call for signature schemes with efficient verification algorithms (e.g., [19]), especially when the client who is verifying signatures is less efficient that the server who generates them. Our result indicates that there is little hope of improving current schemes based simply on one-way permutations.

1.3 Open Problems

Our lower bound for encryption matches the known upper bound for schemes semantically secure against chosen plaintext attack. Clearly, the lower bound also holds for stronger forms of security (such as non-malleability or security against chosen-ciphertext attacks [4]). Here, however, the bounds are not tight. It is natural to wonder whether our bounds can be improved; i.e., whether one can show that these stronger notions of security are intrinsically more difficult to achieve.

In this paper we showed a lower bound for the efficiency of the *verification* algorithm of a signature scheme. An interesting open problem is whether one can find a bound on the number of total invocations made by the signing and key generation algorithms.

Finally, as in [16, 6] our results assume black-box access to one-way (or trapdoor) *permutations*. Since permutations are also functions, our lower bounds apply to the case of one-way (or trapdoor) functions as well. However, in that case the bounds are no longer tight (this is true for the bounds in [6] as well). Constructions based on functions rather than permutation are typically more expensive. Thus it would interesting to see if constructions based on functions can be improved or if stronger lower bounds can be proven in that case.

2 The Models

2.1 Model and Technical Lemmas for One-Way Permutations

Following prior work, we model black-box access to a one-way permutation by giving algorithms access to an oracle π for a *random* permutation, which is one-way with high probability. More formally, a function $f : \{0,1\}^n \to \{0,1\}^n$ is (S, ε)-one way if for every circuit A of size $\leq S$ we have

$$\mathbf{Pr}_x[A(f(x)) \in f^{-1}(f(x))] \leq \varepsilon.$$

(To reduce the number of parameters, we will also say that a function is S*-one way* or *one-way with security* S if it is $(S, 1/S)$-one way. When clear from the context, we may also say the function is S*-hard.*) Denote by Π_t the set of all permutations over $\{0,1\}^t$. The following result then holds:

Theorem 1 ([6]) *For sufficiently large t, if we pick at random an oracle $\pi \in \Pi_t$, then with probability at least $1 - 2^{-2^{t/2}}$ the oracle is $2^{t/5}$-one way.*

For $t < n$, denote by $\Pi_{t,n}$ the subset of Π_n such that $\pi \in \Pi_{t,n}$ iff $\pi(a,b) = (\hat{\pi}(a), b)$ for some $\hat{\pi} \in \Pi_t$. A corollary to Theorem 1 is that if $t = 5 \log S_p$, then for any $n > t$, a randomly chosen $\pi \in \Pi_{t,n}$ is S_p-one-way with very high probability.

Corollary 1 *For sufficiently large t, for any $n > t$, if we pick at random an oracle $\pi \in \Pi_{t,n}$, then with probability at least $1 - 2^{-2^{t/2}}$ the oracle is $2^{t/5}$-one way.*

2.2 Model and Technical Lemmas for Trapdoor Permutations

The model for *trapdoor* permutations is somewhat more involved. We represent a family of trapdoor permutations as a tri-partite oracle $\tau = (G, F, F^{-1})$. Informally, G will be the *key generation* oracle which when queried on a string tk (intended as a "trapdoor") produces the corresponding public key k. The oracle F is the actual trapdoor permutation, which will be queried on key k and an input x. The oracle F^{-1} allows inversion of F; i.e., if $G(tk) = k$ and $F(k, x) = y$, then $F^{-1}(tk, y) = x$.

More formally, we consider a family $T_n = \{\tau \mid \tau = (G, F, F^{-1})\}$ of oracles where:

- $G \in \Pi_n$ is an oracle for a random permutation over $\{0,1\}^n$.
- $F : \{0,1\}^{2n} \to \{0,1\}^n$ is an oracle such that, for each $k \in \{0,1\}^n$, $F(k, \cdot)$ is a random permutation on $\{0,1\}^n$ (independent of $F(k', \cdot)$ for $k' \neq k$).
- $F^{-1} : \{0,1\}^{2n} \to \{0,1\}^n$ is an oracle defined as follows: $F^{-1}(tk, y)$ returns the unique x such that $G(tk) = k$ and $F(k, x) = y$.

We say that trapdoor permutation family $\tau = (G, F, F^{-1})$ is (S, ε)-trapdoor one way if for every circuit A of size $\leq S$ we have

$$\mathop{\mathbf{Pr}}_{x,tk}[k := G(tk) \ : \ A(k, F(k,x)) = x] \leq \varepsilon.$$

As before, we say that a permutation is *S-trapdoor one way* or *trapdoor one-way with security S* if it is $(S, 1/S)$-trapdoor one way. When clear from the context, we may also say the permutation is *S-hard*.

We now show an analogue of Theorem 1 for trapdoor permutations. The basic idea is to notice that, for any k, $F(k, \cdot)$ is a random permutation so Theorem 1 applies except that we need to take into account the possibility that the "adversary" is able to invert G. But since G is also a random permutation, all we need to do is apply Theorem 1 twice. The proof of the following Theorem appears in Appendix A.

Theorem 2 *For sufficiently large t, if we pick at random an oracle $\tau \in T_t$, then with probability at least $1 - 2^{1-2^{t/2}}$ the oracle is $2^{t/6}$-trapdoor one-way.*

Consider an oracle $\hat{\tau} = (\hat{G}, \hat{F}, \hat{F}^{-1}) \in T_t$. From $\hat{\tau}$ we can construct an oracle $\tau = (G, F, F^{-1}) \in T_n$ as:[1]

- $G(tk) = G(tk_a, tk_b) = (\hat{G}(tk_a), tk_b)$. That is, $G \in \Pi_{t,n}$ is a permutation acting only on the first t bits of its input.
- $F(k, x) = F((k_a, k_b), (x_a, x_b)) = (\hat{F}(k_a, x_a), x_b)$. That is, $F(k, \cdot) \in \Pi_{t,n}$ is a permutation acting only on the first t bits of its input. Moreover, this permutation is uniquely determined by the first t bits of k.

[1] We use the following notation: if $|x| = n$, we write $x = (x_a, x_b)$ with $|x_a| = t$ and $|x_b| = n - t$.

- $F^{-1}(tk, y)$ returns the unique x such that $G(tk) = k$ and $F(k, x) = y$.

For $t < n$, we define $T_{t,n} \subset T_n$ as the family of oracles so constructed, for all $\hat{\tau} \in T_t$. It is not hard to see that "inverting" an oracle $\tau \in T_{t,n}$ requires "inverting" the underlying $\hat{\tau} \in T_t$. Thus, we obtain the following:

Corollary 2 *For sufficiently large t, for any $n > t$, if we pick at random an oracle $\tau \in T_{t,n}$, then with probability at least $1 - 2^{1-2^{t/2}}$ the oracle is $2^{t/6}$-trapdoor one-way.*

2.3 Model for Constructions of Encryption Schemes

Public-Key Encryption. A construction of a public-key encryption scheme for m-bit messages is a procedure $\mathcal{PKE}^{(\cdot)}$ that is given access to an oracle $\tau \in T_n$ for a (family of) trapdoor permutations. $\mathcal{PKE}^{(\cdot)}$ consists of the following algorithms:

- *Key generation algorithm* $\mathsf{Gen}^{(\cdot)}$ is a probabilistic algorithm which generates a key pair (pk, sk). We say pk is the public key and sk is the secret key.
- *Encryption algorithm* $\mathsf{Enc}^{(\cdot)}$ is a probabilistic algorithm which, on input a public key pk and a message $M \in \{0,1\}^m$, outputs a ciphertext C.
- *Decryption algorithm* $\mathsf{Dec}^{(\cdot)}$ is a deterministic algorithm which, on input the secret key sk and a ciphertext C, outputs either a message $M \in \{0,1\}^m$ or $\perp$.

We assume that $\mathcal{PKE}^{(\cdot)}$ is *correct*; that is, for all $\tau \in T_n$, for all (pk, sk) output by Gen^τ, for all $M \in \{0,1\}^m$, and for all C output by $\mathsf{Enc}^\tau(pk, M)$ we have $\mathsf{Dec}^\tau(sk, C) = M$. Our results can be modified appropriately if this is not the case.

We now define the security of a given construction $\mathcal{PKE}^{(\cdot)}$. We say two random variables $\mathcal{X}^\tau, \mathcal{Y}^\tau$ which depend upon an oracle τ are (S, ε)-indistinguishable (we will write $\mathcal{X} \overset{(S,\varepsilon)}{\approx} \mathcal{Y}$), if for every circuit Dist of size $\leq S$ we have

$$|\Pr_{x \in \mathcal{X}^\tau}[\mathsf{Dist}^\tau(x) = 1] - \Pr_{x \in \mathcal{Y}^\tau}[\mathsf{Dist}^\tau(x) = 1]| \leq \varepsilon.$$

For a given τ and any message $M \in \{0,1\}^m$ we denote by $\mathcal{PKE}^\tau(M)$ the probability distribution over the *entire* view of an adversary eavesdropping on communication to the receiver; i.e.,

$$\mathcal{PKE}^\tau(M) \stackrel{\text{def}}{=} \{(pk, sk) \leftarrow \mathsf{Gen}^\tau; C \leftarrow \mathsf{Enc}^\tau(pk, M) : (pk, C)\}.$$

We say that $\mathcal{PKE}^{(\cdot)}$ is an $(S_{tp}, S_e, \varepsilon)$-secure PKE construction if for every oracle $\tau \in T_n$ that is S_{tp}-trapdoor one way and for all $M_0, M_1 \in \{0,1\}^m$, we have

$$\mathcal{PKE}^\tau(M_0) \overset{(S_e,\varepsilon)}{\approx} \mathcal{PKE}^\tau(M_1).$$

Private-Key Encryption. A similar model can be defined for private-key encryption. See Appendix B for a detailed description.

2.4 Model for Constructions of Signature Schemes

A construction of a digital signature scheme for m-bit messages is a procedure $\mathcal{SIG}^{(\cdot)}$ that is given access to an oracle $\pi \in \Pi_n$ for a one-way permutation. $\mathcal{SIG}^{(\cdot)}$ consists of:

- *Key generation algorithm* $\mathsf{Gen}^{(\cdot)}$ is a probabilistic algorithm which generates a key pair (PK, SK). We say that PK is the public key and SK is the secret key.
- *Signing algorithm* $\mathsf{Sign}^{(\cdot)}$ is a probabilistic algorithm which, on input SK and a message $M \in \{0,1\}^m$, outputs a signature σ.
- *Verification algorithm* $\mathsf{Vrfy}^{(\cdot)}$ is a deterministic algorithm which, on input PK, a message M, and a signature σ, outputs a single bit.

We assume that $\mathcal{SIG}^{(\cdot)}$ is *correct*; i.e., for all $\pi \in \Pi_n$, for all (PK, SK) output by Gen^π, for all $M \in \{0,1\}^m$, and for all σ output by $\mathsf{Sign}^\pi(SK, M)$ we have $\mathsf{Vrfy}^\pi(PK, M, \sigma) = 1$.

Our definition of a secure signature scheme is very weak: we only require the scheme to be secure against an adversary who sees a signature on a *single* random message (i.e., we consider a *one-time signature scheme secure under random message attack*). Our lower bounds apply even to "weakly-secure" schemes of this type. Since any signature scheme secure against adaptive chosen-message attack (cf. [12]) trivially achieves this "weak" level of security, our results are stronger and immediately imply a bound for this case.

We say that $\mathcal{SIG}^{(\cdot)}$ is an $(S_p, S_\Sigma, \varepsilon)$ signature scheme construction if for every $\pi \in \Pi_n$ that is S_p-one way and for all circuits A of size $\leq S_\Sigma$ we have

$$\Pr\left[\begin{array}{l} (PK, SK) \leftarrow \mathsf{Gen}^\pi; M \leftarrow \{0,1\}^m; \sigma \leftarrow \mathsf{Sign}^\pi(SK, M); \\ (M', \sigma') := A^\pi(PK, M, \sigma) : \mathsf{Vrfy}^\pi(PK, M', \sigma') = 1 \wedge M' \neq M \end{array}\right] \leq \varepsilon.$$

3 Lower Bounds For Public- and Private-Key Encryption

3.1 Public-Key Encryption

Let $\mathcal{PKE}^{(\cdot)} = (\mathsf{Gen}^{(\cdot)}, \mathsf{Enc}^{(\cdot)}, \mathsf{Dec}^{(\cdot)})$ be a public-key encryption scheme for messages of length m that expects as an oracle a (family of) trapdoor permutations τ, and suppose $\mathcal{PKE}^\tau$ is secure whenever τ is S_{tp}-hard. We prove that unless Enc^τ queries τ at least $\Omega(m/\log S_{tp})$ times, an unconditional one-way function exists. In more detail, we first show that unless Enc^τ makes at least $\Omega(m/\log S_{tp})$ queries to τ we can explicitly construct an interactive, private-key encryption scheme $(\mathsf{Enc}', \mathsf{Dec}')$ requiring no access to the oracle and in which the encrypted message is longer than the shared key. Previous results of Impagliazzo and Luby [14] show that this implies the existence of a one-way function.

Our construction of a private-key encryption scheme proceeds by noticing that τ is S_{tp}-hard even when $G : \{0,1\}^n \to \{0,1\}^n$ randomly permutes the first $t = 6 \log S_{tp}$ bits of its input and leaves the remaining bits fixed, and furthermore $F(k, \cdot) : \{0,1\}^n \to \{0,1\}^n$ (for each $k \in \{0,1\}^n$) randomly permutes the first t bits of its input and leaves the remaining bits fixed (i.e., $\tau \in T_{t,n}$; see Corollary 2). Thus τ can be "simulated", and queries by Gen, Enc, and Dec to τ can be "answered", by generating t bits of randomness to prepare the answers as needed (a more detailed description of the simulation of τ is given in the proof of Theorem 4). This suggests a 2-round encryption scheme for m-bit messages (where both parties simulate τ as described) in which the receiver first runs Gen to generate (pk, sk) and sends pk to the sender. The sender computes $C \leftarrow \mathsf{Enc}(pk, M)$ and sends C back to the receiver. The receiver computes M via $\mathsf{Dec}(sk, C)$.

While this encryption scheme is "secure", it may not be correct! Indeed, to ensure correct decryption the parties need to make sure that their simulated calls to τ are answered consistently. Clearly, this can be achieved by having each party send, along with his message, a list of t-prefixes for the queries and answers generated thus far. Now, however, security is no longer guaranteed. But this can be easily remedied: the parties share a random one-time pad in advance and then "encrypt" their list of queries and answers using this pad.

Let q_g be the number of queries made to τ by Gen, and let q_e be the number of queries made by Enc. We show that the private-key encryption scheme outlined above requires a shared key of size $5t \cdot (q_g + q_e)$ to encrypt an m-bit message. Recalling the result of Impagliazzo and Luby [14], this construction requires that $q_g + q_e > m/5t$ since the opposite would imply the unconditional existence of a one-way function. This already gives a "weak" lower bound. To obtain the better lower bound $q_e > m/5t$, an additional trick is needed; details are given in the proof of Theorem 4.

We begin by stating the result of Impagliazzo and Luby that the existence of a private-key encryption scheme $(\mathsf{Enc}', \mathsf{Dec}')$ which securely encrypts messages longer than the shared key implies the existence of a one-way function. We in fact give a precise statement of their results as applicable to the present context; a proof appears in Appendix C. Note that the Theorem applies even to an interactive encryption protocol.

Theorem 3 ([14]) *Let* $(\mathsf{Enc}, \mathsf{Dec})$ *be an* (S, δ)*-secure (possibly interactive), private-key encryption scheme in which the shared key is of length* k *and the encrypted message is of length* $m > k$*. Let* S_e *be the size of the circuit for* Enc*. Then there exists an* $(S - S_e, \delta + 2^{k-m})$*-one-way function.*

Our main result is the following:

Theorem 4 *Let* $(\mathsf{Gen}^\tau, \mathsf{Enc}^\tau, \mathsf{Dec}^\tau)$ *be an* $(S_{tp}, S_e, \varepsilon)$*-secure PKE construction for messages of length* m *in which* Gen *makes* q_g *queries to trapdoor permutation family* τ*,* Enc *makes* $q_e < m/30 \log S_{tp}$ *queries to* τ*, and* $\varepsilon < (1/2 - 2^{1-S_{tp}}) \cdot (2 \cdot \lceil q_g/(m/30 \log S_{tp} - q_e) \rceil)^{-1}$*. Then there exists a one-way function that requires no access to* τ*.*

The proof appears in Appendix D.

3.2 Private-Key Encryption

The techniques of the previous section can be adapted to show a similar lower bound for private-key encryption schemes. Specifically, let $\mathcal{SKE}^{(\cdot)} = (\mathsf{Enc}^{(\cdot)}, \mathsf{Dec}^{(\cdot)})$ be a private-key encryption scheme for m bit messages in which the shared key is of length k, and suppose $\mathcal{SKE}^{\pi}$ is secure whenever π is S_p-hard. We show that unless Enc^{π} queries π at least $q_e = \Omega(\frac{m-k}{\log S})$ times, then an unconditional one-way function exists. This matches the known upper bound.

The proof is similar to that of Theorem 4, in that we convert $\mathcal{SKE}^{(\cdot)}$ to a private-key encryption scheme $\mathcal{SKE}'$ that does not access the oracle at all. The only difference between the proof here and the proof of Theorem 4 is that here the parties need to share a k-bit key in addition to the one-time pad used to encrypt their "simulated" queries and answers to the oracle. Set $t = 5 \log S_p$. The resulting $\mathcal{SKE}'$ requires a shared key of length $k + 2tq_e$ and encrypts an m-bit message. As before, then, this requires that $k + 2tq_e > m$ since the opposite implies the unconditional existence of a one-way function (cf. Theorem 3).

Theorem 5 *Let* $(\mathsf{Enc}^{\pi}, \mathsf{Dec}^{\pi})$ *be an* (S_p, S_e, ε)*-secure SKE construction for messages of length* m *in which* Enc *makes* $q_e < \frac{m-k}{10 \log S_p}$ *queries to a permutation* π*, and* $\varepsilon < 1/2 - 2^{-S_p}$*. Then there exists a one-way function that requires no access to* π*.*

4 Lower Bound for Signature Schemes

We now show a lower bound on the efficiency of black-box constructions for signature schemes based on one-way permutations (we no longer consider trapdoor permutations since one-way permutations suffice to construct signature schemes [17, 18, 20]). Let $\mathcal{SIG}^{(\cdot)} = (\mathsf{Gen}^{(\cdot)}, \mathsf{Sign}^{(\cdot)}, \mathsf{Vrfy}^{(\cdot)})$ be a signature scheme for messages of length m that expects as an oracle a permutation π, and suppose that $\mathcal{SIG}^{\pi}$ is secure whenever π is S_p-hard. We prove that unless Vrfy queries π at least $\Omega(m/\log S_p)$ times, then it is possible to construct from $\mathcal{SIG}$ an unconditional one-way function. Of course, this one-way function can then be used to construct an unconditionally-secure signature scheme [17, 18, 20] (i.e., without any access to π).

We give an informal overview of our proof technique (formalized in the proof of Theorem 6). As a first attempt to construct a one-way function, one might define

$$F_1(PK, M, \sigma) = PK \| \mathsf{Vrfy}^{(\cdot)}(PK, M, \sigma).$$

Intuitively, this function is difficult to invert on elements of the form $PK\|1$ if PK is a valid and randomly-chosen public key, since inverting the function on points of this type amounts to signature forgery. As presently defined, however, evaluating F_1 requires calls to π and is therefore not unconditionally one-way. As in Section 3, however, one may observe that π is S_p-hard (and thus $\mathcal{SIG}$ is secure) when π randomly permutes the first $t = 5 \log S_p$ bits of its input while leaving the remaining bits fixed (cf. Corollary 1). If Vrfy makes q queries to its oracle for π, then specifying $q\,t$ bits as the answers to these queries removes any need to query the oracle. Based on this approach, one might consider the function

$$F_2(PK, y_1, \ldots, y_q, M, \sigma) = PK \| y_1 \| \cdots \| y_q \| x_1 \| \cdots \| x_q \| \mathsf{Vrfy}^{y_1,\ldots,y_q}(PK, M, \sigma), \tag{1}$$

where $|y_1| = \cdots = |y_q| = |x_1| = \cdots = |x_q| = t$ and the i^{th} query $x_i \| b_i$ of Vrfy is answered with $y_i \| b_i$ (as in Section 3, we may assume without loss of generality that Vrfy queries π with strings having distinct t-prefixes). The intuition, as before, is that F_2 is difficult to invert on elements of the form $PK \| \vec{y} \| \vec{x} \| 1$ if $PK, \vec{y}$, and $\vec{x}$ are chosen appropriately.

Now, however, another problem arises. In order for F_2 to qualify as a one-way function, it must be hard to invert $F_2(PK, \vec{y}, M, \sigma)$ when $PK, \vec{y}, M, \sigma$ are sampled from an *(efficiently) sampleable distribution* (cf. Lemma 2 in Appendix E). More specifically, a proof of one-wayness will need to show how to efficiently sample $PK, \vec{y}, M, \sigma$ such that inverting $F_2(PK, \vec{y}, M, \sigma)$ results in a forged signature and hence a contradiction. A necessary condition for inversion to result in signature forgery is that $\mathsf{Vrfy}^{\vec{y}}(PK, M, \sigma) = 1$. Generating $PK, \vec{y}, M, \sigma$ such that this holds is easy if we have the secret key SK; but in this case, inverting F_2 and forging a signature does not yield the desired contradiction! Similarly, if M, σ are obtained from the signer, then inverting $F_2(PK, \vec{y}, M, \sigma)$ does not necessarily result in a forgery if inversion yields the same message/signature pair.

We come now to the crux of our proof. Set $Y = PK \| \vec{y} \| \vec{x} \| 1 = F_2(PK, \vec{y}, M, \sigma)$. Informally, as long as there is an element $PK \| \vec{y} \| M' \| \sigma' \in F_2^{-1}(Y)$ with $M' \neq M$, then inverting Y will result in a forgery with noticeable probability. Furthermore (and completing the proof sketch), there exists such an element with sufficiently high probability (over choice of Y) exactly when $|\vec{x}| = |x_1| + \cdots + |x_q| < |M|$, giving the desired bound.

Theorem 6 *Let $\mathcal{SIG}^\pi = (\mathsf{Gen}^\pi, \mathsf{Sign}^\pi, \mathsf{Vrfy}^\pi$ be an $(S_p, S_\Sigma, \varepsilon)$-secure signature scheme construction for messages of length m in which $\varepsilon < 1/2 - S_p^{-1}$ and Vrfy makes $q < m/5 \log S_p$ queries to a permutation oracle π. Then there exists a one-way function that requires no access to π.*

References

[1] M. Bellare, S. Halevi, A. Sahai, and S. Vadhan. Many-to-one Trapdoor Functions and their relations to Public-Key Cryptosystems. Crypto '98.

[2] M. Bellare and P. Rogaway. Collision-Resistant Hashing: Toward Making UOWHFs Practical. Crypto '97.

[3] M. Blum and S. Micali. How to Generate Cryptographically Strong Sequences of Pseudo-Random Bits. *SIAM J. Comp.*, 13(4): 850–864 (1984).

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

[5] T. El Gamal. A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. *IEEE Transactions on Information Theory* 31(4): 469–472 (1985).

[6] R. Gennaro and L. Trevisan. Lower Bounds on the Efficiency of Generic Cryptographic Constructions. FOCS 2000.

[7] Y. Gertner, S. Kannan, T. Malkin, O. Reingold, and M. Viswanathan. The Relationship between Public-Key Encryption and Oblivious Transfer. FOCS 2000.

[8] Y. Gertner, T. Malkin, and O. Reingold. On the Impossibility of Basing Trapdoor Functions on Trapdoor Predicates. FOCS 2001.

[9] O. Goldreich. *Foundations of Cryptography: Basic Tools*, Cambridge University Press, 2001.

[10] O. Goldreich and L. Levin. A hard-core predicate for all one-way functions. STOC '89.

[11] S. Goldwasser and S. Micali. Probabilistic Encryption. *J. Computer and System Sciences* 28(2): 270–299 (1984).

[12] S. Goldwasser, S. Micali, and R. Rivest. A Digital Signature Scheme Secure Against Adaptive Chosen Message Attacks. *SIAM J. Computing*, 17(2): 281–308 (1988).

[13] J. Kahn, M. Saks, and C. Smyth. A Dual Version of Reimer's Inequality and a Proof of Rudich's Conjecture. In *Proceedings of the 15th IEEE Conference on Computational Complexity*, 2000.

[14] R. Impagliazzo and M. Luby. One-Way Functions are Essential for Complexity-Based Cryptography. FOCS '89.

[15] R. Impagliazzo and S. Rudich. Limits on the Provable Consequences of One-Way Permutations. STOC '89.

[16] J. H. Kim, D. Simon, and P. Tetali. Limits on the Efficiency of One-Way Permutation-Based Hash Functions. FOCS '99.

[17] L. Lamport. Constructing Digital Signatures from a One-Way Function. Technical Report CSL-98, SRI International, 1979.

[18] M. Naor and M. Yung. Universal One-Way Hash Functions and Their Cryptographic Applications. STOC '89.

[19] A. Perrig. The BiBa One-Time Signature and Broadcast Authentication Protocol. ACM CCCS '01.

[20] J. Rompel. One-Way Functions are Necessary and Sufficient for Digital Signatures. STOC '90.

[21] S. Rudich. *Limits on the provable consequences of one-way functions.* PhD thesis, University of California at Berkeley, 1988.

[22] S. Rudich. The Use of Interaction in Public Cryptosystems. Crypto '91.

[23] D. Simon. Finding Collisions on a One-Way Street: Can Secure Hash Functions be Based on General assumptions? Eurocrypt '98.

[24] A. C. Yao. Theory and applications of trapdoor functions. FOCS '82.

A Proof of Trapdoor Permutation Technical Lemma

Proof We actually show the stronger result that τ is $(2^{t/5}, 2^{1-t/5})$-trapdoor one-way. The simpler expression $2^{t/6}$ follows by trivial upper bounding.

Let A be a circuit of size $S = 2^{t/5}$. From Theorem 1 we know that with probability at least $p = 1 - 2^{-2^{t/2}}$ the permutation G is one-way with security $2^{t/5}$. Since $F(k, \cdot)$ is a random permutation for any k, Theorem 1 also guarantees that with probability at least $p = 1 - 2^{-2^{t/2}}$ the permutation $F(k, \cdot)$ is one-way with security $2^{t/5}$. This holds as long as A^τ never queries the oracle F^{-1} on tk such that $G(tk) = k$. Thus with probability at least $p^2 > 1 - 2^{1-2^{t/2}}$ both G and $F(k, \cdot)$ are $2^{t/5}$-one way.

We have:

$$\Pr_{x,tk}[k := G(tk)\ :\ A^\tau(k, F(k,x)) = x] =$$
$$\Pr_{x,tk}[k := G(tk)\ :\ A^\tau(k, F(k,x)) = x \mid A^\tau \text{ never queries } F^{-1}(tk, y) \text{ for any } y] \qquad (2)$$
$$+ \Pr_{x,tk}[k := G(tk)\ :\ A^\tau(k, F(k,x)) = x \mid A^\tau \text{ queries } F^{-1}(tk, y) \text{ for some } y]. \qquad (3)$$

Now, (2) can be upper bounded by $2^{-t/5}$ since we are conditioning on the event that A^τ never queries F^{-1} on the correct tk. Similarly, (3) can be upper bounded by $2^{-t/5}$ since A^τ can be used as a circuit inverting G on k. Summing gives the desired result. ■

B Model for Private-Key Encryption Schemes

The case of private-key encryption is handled via appropriate modification of the above. A construction of a private-key encryption scheme for m-bit messages using k-bit keys is a procedure $\mathcal{SKE}^{(\cdot)}$ that is given access to an oracle $\pi \in \Pi_n$ for a one-way permutation. In contrast to the public-key setting, $\mathcal{SKE}^{(\cdot)}$ consists only of algorithms $\mathsf{Enc}^{(\cdot)}$ and $\mathcal{D}^{(\cdot)}$. The shared key sk is uniformly distributed in $\{0,1\}^k$. Algorithm $\mathsf{Enc}^{(\cdot)}$, which may be probabilistic, takes as input sk and a message $M \in \{0,1\}^m$ and outputs a ciphertext C. Algorithm $\mathcal{D}^{(\cdot)}$ is a deterministic algorithm which takes as input sk and ciphertext C and outputs a message M. As in the public-key case, we assume that $\mathcal{SKE}^{(\cdot)}$ is *correct*; i.e., for all $\pi \in \Pi_n$, for all $sk \in \{0,1\}^k$, for all $M \in \{0,1\}^m$, and for all C output by $\mathsf{Enc}^\pi(sk, M)$ we have $\mathcal{D}^\pi(sk, C) = M$. Our results can be modified appropriately if this is not the case.

For a given π and any message $M \in \{0,1\}^m$ we denote by $\mathcal{SKE}^\pi(M)$ the probability distribution over the view of an adversary eavesdropping on the communication; i.e.,

$$\mathcal{SKE}^\pi(M) \stackrel{\text{def}}{=} \left\{ sk \leftarrow \{0,1\}^k; C \leftarrow \mathsf{Enc}^\pi(sk, M) : C \right\}.$$

Note that the adversary's view does not include sk, whereas in the public-key case the adversary's view included pk.

We say that $\mathcal{SKE}^{(\cdot)}$ is an (S_p, S_e, ε)-secure SKE construction if for every oracle $\pi \in \Pi_n$ that is S_p-one way and for all $M_0, M_1 \in \{0,1\}^m$, we have

$$\mathcal{SKE}^\tau(M_0) \stackrel{(S_e,\varepsilon)}{\approx} \mathcal{SKE}^\tau(M_1).$$

C Proof of Theorem 3

Proof We show a proof of Theorem 3 for the case of non-interactive encryption; extension to the case of interactive encryption is straightforward (following [14]). Define function f as follows: $f(sk, M, \omega) = \mathsf{Enc}(sk, M; \omega) \| M$, where ω represents the random coins used by Enc. We claim that this function is (S', δ')-one way, where $S' = S - S_e$ and $\delta' = \delta + 2^{k-m}$. Assume the contrary. Then there is an algorithm B such that $\mathsf{Succ}_{B,f} \stackrel{\text{def}}{=} \Pr[sk \leftarrow \{0,1\}^k; M \leftarrow \{0,1\}^m; C \leftarrow \mathsf{Enc}(sk, M) : B(C\|M) \in f^{-1}(C\|M)] > \delta'$ and $|B| \leq S'$.

We use B to construct an algorithm A breaking the security of the encryption scheme. Let M_0, M_1 be two messages chosen randomly from $\{0,1\}^m$. On input C (representing either an encryption of M_0 or M_1), algorithm A computes $sk' \| M' \| \omega' := B(C\|M_0)$ and checks whether $f(sk', M', \omega') \stackrel{?}{=} C\|M_0$. If not, then A outputs 0. If so, then A outputs 1. Note that $|A| = |B| + S_e \leq S$.

We first note that $\Pr_{C \in \mathcal{SKE}(M_0)}[A(C) = 1] = \mathsf{Succ}_{B,f} > \delta'$. Secondly, we have

$$\begin{aligned}
&\Pr_{C \in \mathcal{SKE}(M_1)}[A(C) = 1] \\
&\leq \Pr[sk \leftarrow \{0,1\}^k; M_0, M_1 \leftarrow \{0,1\}^m; C \leftarrow \mathsf{Enc}(sk, M_1) : \exists sk' \text{ s.t. } \mathsf{Dec}(sk', C) = M_0] \\
&\leq \sum_{sk'} \Pr[sk \leftarrow \{0,1\}^k; M_0, M_1 \leftarrow \{0,1\}^m; C \leftarrow \mathsf{Enc}(sk, M_1) : \mathsf{Dec}(sk', C) = M_0] \\
&\leq \sum_{sk'} 2^{-m} \\
&= 2^{k-m}.
\end{aligned}$$

Then $\left|\Pr_{C \in \mathcal{SKE}(M_0)}[A(C) = 1] - \Pr_{C \in \mathcal{SKE}(M_1)}[A(C) = 1]\right| > \delta' - 2^{k-m} \geq \delta$, a contradiction. ■

D Proof of Encryption Lower Bound

We introduce notation which will be useful for the proof of the theorem. For any $\ell \in \mathbb{N}$ and any public-key encryption scheme $(\mathsf{Gen}^{(\cdot)}, \mathsf{Enc}^{(\cdot)}, \mathsf{Dec}^{(\cdot)})$ for m-bit messages, we define encryption scheme $(\mathsf{Gen}^{(\cdot)}, \mathsf{Enc}_\ell^{(\cdot)}, \mathsf{Dec}_\ell^{(\cdot)})$ for ℓm-bit messages as follows: $\mathsf{Enc}_\ell^{(\cdot)}(pk, M) \stackrel{\text{def}}{=} \mathsf{Enc}^{(\cdot)}(pk, M_1), \ldots, \mathsf{Enc}^{(\cdot)}(pk, M_\ell)$ (where $M = M_1, \ldots, M_\ell$, and independent random coins are used for each invocation of Enc) and $\mathsf{Dec}_\ell^{(\cdot)}$ is defined in the natural way. The following lemma describes the well-known result [11] that this encryption scheme is also secure.

Lemma 1 *Let $(\mathsf{Gen}^\tau, \mathsf{Enc}^\tau, \mathsf{Dec}^\tau)$ be an $(S_{tp}, S_e, \varepsilon)$-secure PKE construction for m-bit messages and let S_{Enc} be the size of the circuit for Enc. Then $(\mathsf{Gen}^\tau, \mathsf{Enc}_\ell^\tau, \mathsf{Dec}_\ell^\tau)$ (as defined above) is an $(S_{tp}, S_e - (\ell-1)S_{\mathsf{Enc}}, \ell\varepsilon)$-secure PKE construction for ℓm-bit messages.*

We can now prove Theorem 4.

Proof Let $t = 6 \log S_{tp}$ and set $\ell = 2 \cdot \lceil 5t\, q_g/(m - 5t\, q_e) \rceil$. Lemma 1 shows that encryption scheme $\mathcal{PKE}_\ell^{(\cdot)}$, as defined above, is an $(S_{tp}, S', \ell\varepsilon)$-secure PKE construction for ℓm-bit messages, where $S' = S_e - (\ell-1)S_{\mathsf{Enc}}$. The number of queries made by Enc_ℓ to its oracle is at most ℓq_e. We use $\mathcal{PKE}_\ell^{(\cdot)}$ to construct an (S', δ)-secure interactive, private-key encryption scheme $\mathcal{SKE} = (\mathsf{Enc}', \mathsf{Dec}')$, where $\delta = \ell\varepsilon + 2^{1-S_{tp}}$. Furthermore, the shared key for this scheme will be of length $5t \cdot (q_g + \ell q_e)$ and the encrypted message will be of length ℓm. Finally, $\mathcal{SKE}$ will require no access to the trapdoor permutation oracle. Since $5t \cdot (q_g + \ell q_e) < 5t(\ell m/5t) < \ell m$ and $\delta < 1/2$, Theorem 3 yields the statement of the theorem.

Since $\mathcal{PKE}_\ell^{(\cdot)}$ is an $(S_{tp}, S', \ell\varepsilon)$-secure public-key encryption scheme, it means that if τ is S_{tp}-hard then, for any circuit B of size $\leq S'$ and for any messages $M_0, M_1 \in \{0,1\}^{\ell m}$ we have

$$\left| \Pr_{v \in \mathcal{PKE}_\ell^\tau(M_0)}[B^\tau(v) = 1] - \Pr_{v \in \mathcal{PKE}_\ell^\tau(M_1)}[B^\tau(v) = 1] \right| \leq \ell\varepsilon.$$

Corollary 2 indicates that a random $\tau \in T_{t,n}$ is S_{tp}-hard except with probability at most $2^{1-S_{tp}}$. A straightforward averaging argument shows that for any circuit B of size $\leq S'$ and for any messages $M_0, M_1 \in \{0,1\}^{\ell m}$ we have

$$\left| \Pr_{\tau \in T_{t,n};\, v \in \mathcal{PKE}_\ell^\tau(M_0)}[B^\tau(v) = 1] - \Pr_{\tau \in T_{t,n};\, v \in \mathcal{PKE}_\ell^\tau(M_1)}[B^\tau(v) = 1] \right| \leq \ell\varepsilon + 2^{1-S_{tp}} = \delta < 1/2.$$

As mentioned in the discussion of Section 3.1, our private-key encryption scheme $\mathcal{SKE}$ will need to "simulate" a random $\tau \in T_{t,n}$ for algorithms $\mathsf{Gen}, \mathsf{Enc}_\ell$, and Dec_ℓ. Before describing $\mathcal{SKE}$, then, we first describe the details of this simulation. The simulation procedure $\mathcal{SIM}$ must ensure consistency of the answers to all oracle queries. It does so by storing t-prefixes of all previous oracle queries and answers in a list L. Before answering the current query, $\mathcal{SIM}$ ensures that the current answer will not generate any inconsistencies. For example, if query $tk\|b$ to G was answered by $k\|b$ (where $|tk| = |k| = t$), then subsequent query $tk'\|b'$ to G must be answered by $k'\|b'$ where $k' = k$ iff $tk' = tk$. A more involved procedure is needed to answer queries to F and F^{-1}.

Procedure 1 $\mathcal{SIM}(L)$

- *On query* $tk\|b$ *to* G *(where* $|tk| = t$*): if* $\exists k$ *s.t.* $(tk, k) \in L$*, return* $k\|b$*. Otherwise pick random* $k \in \{0,1\}^t$ *such that* $\forall tk' : (tk', k) \notin L$*, return* $k\|b$*, and store* (tk, k) *in* L*.*
- *On query* $(k\|b, x\|b')$ *to* F *(where* $|k| = |x| = t$*):*
  1. *if* $\exists y$ *s.t.* $(k, x, y) \in L$*, return* $y\|b'$*.*
  2. *Otherwise, if* $\exists tk$ *s.t.* $(tk, k) \in L$*, choose random* $y \in \{0,1\}^t$ *such that* $\forall x' : (k, x', y) \notin L$*, return* $y\|b'$*, and store* (k, x, y) *in* L*.*
  3. *Otherwise, choose random* $tk \in \{0,1\}^t$ *such that* $\forall k' : (tk, k') \notin L$*, choose random* $y \in \{0,1\}^t$*, return* $y\|b'$*, and store* (k, tk) *and* (k, x, y) *in* L*.*
- *On query* $(tk\|b, y\|b')$ *to* F^{-1} *(where* $|tk| = |y| = t$*):*
  1. *if* $\exists k, x$ *s.t.* $(tk, k), (k, x, y) \in L$*, return* $x\|b'$
  2. *Otherwise, if* $\exists k$ *s.t.* $(tk, k) \in L$*, choose random* $x \in \{0,1\}^t$ *such that* $\forall y' : (k, x, y') \notin L$*, return* $x\|b'$*, and store* (k, x, y) *in* L
  3. *Otherwise, choose random* $k \in \{0,1\}^t$ *such that* $\forall tk' : (tk', k) \notin L$*, choose random* $x \in \{0,1\}^t$*, return* $x\|b'$*, and store* (tk, k) *and* (k, x, y) *in* L

Note that each time a query is answered, at most $5t$ bits are stored in L.

We construct $\mathcal{SKE}$ as follows. Let the shared key be $s = (s_1, s_2)$ where $|s_1| = 5t\, q_g$ and $|s_2| = 5t\, \ell q_e$. To encrypt message M, the receiver Dec' begins by initializing list $L := \emptyset$. The receiver then computes $(pk, sk) \leftarrow \mathsf{Gen}^{\mathcal{SIM}(L)}$ and sends $pk, s_1 \oplus L$ to the sender Enc'. The receiver stores sk, L for later use. Upon receiving the first message $pk, \hat{s}_1$, the sender computes $L_1 := s_1 \oplus \hat{s}_1$ and sets $L := L_1$. The sender then computes $C \leftarrow \mathsf{Enc}_\ell^{\mathcal{SIM}(L)}(pk, M)$ and sets $L_2 := L \backslash L_1$. Finally, Enc' sends $C, s_2 \oplus L_2$ to the receiver. Upon receiving message $C, \hat{s}_2$, the receiver decrypts by setting $L_2 := s_2 \oplus \hat{s}_2$ and $L := L \cup L_2$ (recall L is the stored list from the first stage). The receiver can then compute $M := \mathsf{Dec}_\ell^{\mathcal{SIM}(L)}(sk, C_i)$.

Because $\mathcal{SIM}$ perfectly simulates a random $\tau \in T_{t,n}$, it is clear that $\mathcal{SKE}$ has correct decryption. We now show that the scheme is (S', δ)-secure. Assume toward a contradiction that there exists a circuit A of size $\leq S'$ and messages $M_0, M_1 \in \{0,1\}^{\ell m}$ such that

$$\left| \mathop{\mathbf{Pr}}_{v \in \mathcal{SKE}(M_0)}[A(v) = 1] - \mathop{\mathbf{Pr}}_{v \in \mathcal{SKE}(M_1)}[A(v) = 1] \right| > \delta.$$

We construct circuit B breaking the security of $\mathcal{PKE}_\ell$ as follows. On input pk, C, circuit B picks random strings $\hat{s}_1, \hat{s}_2$ where $|\hat{s}_1| = 5t\, q_g$ and $|\hat{s}_2| = 5t\, \ell q_e$. Then, B outputs $A(pk, \hat{s}_1, C, \hat{s}_2)$. Note that B makes no use of the oracle τ. Because the keys s_1, s_2 of $\mathcal{SKE}$ are used as a one-time pad, it is easy to see that:

$$\left| \mathop{\mathbf{Pr}}_{\tau \in T_{t,n};\ (pk, C) \in \mathcal{PKE}_\ell^\tau(M_0)}[B(pk, C) = 1] - \mathop{\mathbf{Pr}}_{\tau \in T_{t,n};\ (pk, C) \in \mathcal{PKE}_\ell^\tau(M_1)}[B(pk, C) = 1] \right|$$

$$\begin{aligned} &= \left| \Pr_{v \in \mathcal{SKE}(M_0)}[A(pk, \hat{s}_1, C, \hat{s}_2) = 1] - \Pr_{v \in \mathcal{SKE}(M_1)}[A(pk, \hat{s}_1, C, \hat{s}_2) = 1] \right| \\ &> \delta, \end{aligned}$$

a contradiction. ■

E Proof of Signature Lower Bound

Before giving the details of the proof, we present a technical lemma showing that the existence of a function which is one-way over an (efficiently) sampleable domain implies the existence of a function which is one-way under the standard definition.

Lemma 2 *Let $\mathcal{D}$ be a distribution computable by a circuit of size $S_{\mathcal{D}}$ and let f be a function such that for every circuit A of size $\leq S$ we have:* $\Pr[x \leftarrow \mathcal{D} : A(f(x)) \in f^{-1}(f(x))] \leq \delta$. *Then there exists a function f' that is $(S - S_{\mathcal{D}}, \delta)$-one way.*

Proof Define $f'(r) \stackrel{\text{def}}{=} f(\mathcal{D}(r))$. We claim that f' is $(S - S_{\mathcal{D}}, \delta)$-one way. Assume the contrary. Then there exists a circuit A' of size at most $S - S_{\mathcal{D}}$ for which

$$\Pr_r[A'(f'(r)) \in f'^{-1}(f'(r))] > \delta.$$

Define circuit A by $A(y) \stackrel{\text{def}}{=} \mathcal{D}(A'(y))$. Notice that $|A| \leq S$. Furthermore,

$$\begin{aligned} \Pr[x \leftarrow \mathcal{D} : A(f(x)) \in f^{-1}(f(x))] &= \Pr_r[x := \mathcal{D}(r) : f(A(f(x))) = f(x)] \\ &= \Pr_r[f(A(f'(r))) = f'(r)] \\ &= \Pr_r[f'(A'(f'(r))) = f'(r)] \\ &= \Pr_r[A'(f'(r)) \in f'^{-1}(f'(r))] \\ &> \delta, \end{aligned}$$

a contradiction. ■

We are now ready for the proof of Theorem 6.

Proof Let S_{Vrfy} be the size of the circuit for Vrfy. Our proof proceeds by using $\mathcal{SIG}$ to construct a function F along with a distribution $\mathcal{D}$ such that for every circuit A of size $\leq S_{\Sigma} - S_{\mathsf{Vrfy}}$ we have:

$$\Pr[X \leftarrow \mathcal{D} : A(F(X)) \in F^{-1}(F(X))] \leq \varepsilon + 2^{-S_p} + 1/2 < 1.$$

Lemma 2 then yields the statement of the Theorem.

If $\mathcal{SIG}$ is an $(S_p, S_{\Sigma}, \varepsilon)$ signature scheme, it means that if $\pi : \{0,1\}^n \rightarrow \{0,1\}^n$ is S_p-hard then, for any circuit B of size $\leq S_{\Sigma}$ we have $\Pr[\mathsf{Forge}_{\pi,B}] \leq \varepsilon$ where

$$\begin{aligned} &\mathsf{Forge}_{\pi,B} \stackrel{\text{def}}{=} \\ &\quad (PK, SK) \leftarrow \mathsf{Gen}^{\pi}; M \leftarrow \{0,1\}^m; \sigma \leftarrow \mathsf{Sign}^{\pi}(SK, M); \\ &\quad (M', \sigma') := B^{\pi}(PK, M, \sigma) : \mathsf{Vrfy}^{\pi}(PK, M', \sigma') = 1 \wedge M' \neq M. \end{aligned}$$

Let $t = 5 \log S_p$. Corollary 1 indicates that a random permutation $\pi \in \Pi_{t,n}$ is S_p-hard except with probability at most $2^{-2^{t/2}} < 2^{-S_p}$. An averaging argument then implies that for any circuit B of size $\leq S_\Sigma$ we have $\Pr[\pi \leftarrow \Pi_{t,n} : \mathsf{Forge}_{\pi,B}] < \varepsilon + 2^{-S_p}$.

Our one-way function F is defined as in (1) which we repeat here for convenience:

$$F(PK, \vec{y}, M, \sigma) = PK \| \vec{y} \| \vec{x} \| \mathsf{Vrfy}^{\vec{y}}(PK, M, \sigma),$$

where $\vec{y} = (y_1, \ldots, y_q)$, $\vec{x} = (x_1, \ldots, x_q)$, $|y_1| = \cdots = |y_q| = |x_1| = \cdots = |x_q| = t$, and the i^{th} query $x_i \| b_i$ of Vrfy is answered with $y_i \| b_i$. Define distribution $\mathcal{D}$ as follows: The output of $\mathcal{D}(r_g, r_M, r_s, r_y)$ (where coins $r_g, r_s, r_y \in \{0,1\}^*$ and $r_M \in \{0,1\}^m$ are uniformly distributed) is determined by the following experiment:

$$(PK, SK) := \mathsf{Gen}(r_g); M := r_M; \sigma := \mathsf{Sign}(SK, M; r_s); \mathsf{Vrfy}(PK, M, \sigma) : PK \| \vec{y} \| M \| \sigma,$$

where the random coins r_y are used to "simulate"[2] a random $\pi \in \Pi_{t,n}$, and $\vec{y}$ represents the t-prefixes of the answers given in response to the queries of Vrfy.

Assume toward a contradiction that there exists a circuit A of size $\leq S_\Sigma - S_{\mathsf{Vrfy}}$ for which $\Pr[X \leftarrow \mathcal{D} : A(F(X)) \in F^{-1}(F(X))] > \varepsilon + 2^{-S_p} + 1/2$. We use A to construct an algorithm B that, given PK and a signature σ of a random message M, forges a signature on a new message M'. $B^\pi(PK, M, \sigma)$ first runs $\mathsf{Vrfy}(PK, M, \sigma)$, answering the queries of Vrfy by forwarding them to π. Let $\vec{x}$ be the t-prefixes of the queries made by Vrfy and let $\vec{y}$ be the t-prefixes of the corresponding answers. Define $X = PK \| \vec{y} \| M \| \sigma$ and $Y = PK \| \vec{y} \| \vec{x} \| 1 = F(X)$. B computes $PK' \| \vec{y}' \| M' \| \sigma' = A(Y)$ and outputs (M', σ').

The distribution on X (over random choice of $\pi \in \Pi_{t,n}$) is exactly that specified by $\mathcal{D}$. Furthermore, B outputs a successful forgery whenever (1) $F(PK' \| \vec{y}' \| M' \| \sigma') = Y$ and (2) $M' \neq M$. To see this, note that $F(PK' \| \vec{y}' \| M' \| \sigma') = PK \| \vec{y} \| \vec{x} \| 1$ immediately implies that $\mathsf{Vrfy}^{\vec{y}}(PK, M', \sigma') = 1$ and furthermore Vrfy makes queries with t-prefixes $\vec{x}$. But this then means that $\mathsf{Vrfy}^\pi(PK, M', \sigma') = 1$, since we have already determined that $\pi(x_1 \| b) = y_1 \| b, \ldots, \pi(x_q \| b) = y_q \| b$ (for all b). If furthermore $M' \neq M$, this implies that (M', σ') is a successful forgery. Finally, note that $|B| \leq S_\Sigma$.

Let Equal denote the event that $M' = M$ in the execution of B. Then

$$\begin{aligned}
&\Pr[\pi \leftarrow \Pi_{t,n} : \mathsf{Forge}_{\pi,B}] \\
&\quad \geq \Pr[X \leftarrow \mathcal{D}; Y = F(X); X' = A(Y) : X' \in F^{-1}(Y) \wedge \overline{\mathsf{Equal}}] \\
&\quad = \Pr[X \leftarrow \mathcal{D}; Y = F(X); X' = A(Y) : X' \in F^{-1}(Y)] \\
&\qquad\quad - \Pr[X \leftarrow \mathcal{D}; Y = F(X); X' = A(Y) : X' \in F^{-1}(Y) \wedge \mathsf{Equal}] \\
&\quad > \varepsilon + 2^{-S_p} + 1/2 \\
&\qquad\quad - \Pr[X \leftarrow \mathcal{D};\ PK \| \vec{y} \| \vec{x} \| 1 = F(X);\ PK' \| \vec{y}' \| M' \| \sigma' = A(PK \| \vec{y} \| \vec{x} \| 1) : M' = M] \\
&\quad = \varepsilon + 2^{-S_p} + 1/2
\end{aligned}$$

[2]Simulating a random $\pi \in \Pi_{t,n}$ requires the following: query $x \| b$ is answered with $y \| b$ for random y, except that if query $x' \| b'$ (possibly made by a different algorithm) was previously answered with $y' \| b'$, then $y = y'$ iff $x = x'$ (here, $|x| = |x'| = |y| = |y'| = t$).

$$
\begin{aligned}
& -\sum_{\vec{z}} \Pr[X \leftarrow \mathcal{D};\ PK\|\vec{y}\|\vec{x}\|1 = F(X);\ PK'\|\vec{y}\,'\|M'\|\sigma' = A(PK\|\vec{y}\|\vec{x}\|1) : \\
& \qquad\qquad M' = M \wedge \vec{x} = \vec{z}] \\
\geq\ & \varepsilon + 2^{-S_p} + 1/2 \\
& -\sum_{\vec{z}} \Pr[X \leftarrow \mathcal{D};\ PK\|\vec{y}\|\vec{x}\|1 = F(X);\ PK'\|\vec{y}'\|M'\|\sigma' = A(PK\|\vec{y}\|\vec{z}\|1) : M' = M] \\
=\ & \varepsilon + 2^{-S_p} + 1/2 - \sum_{\vec{z}} 2^{-m}, \qquad (4)
\end{aligned}
$$

where the final equality holds since $M \in \{0,1\}^m$ is chosen uniformly at random independent of PK and $\vec{y}$ (and hence A has no information about M). Noting that the number of terms in the sum of (4) is exactly $2^{|\vec{z}|} \leq 2^{m-1}$, we have:

$$
\Pr[\pi \leftarrow \Pi_{t,n} : \mathsf{Forge}_{\pi,B}] > \varepsilon + 2^{-S_p} + 1/2 - 1/2 = \varepsilon + 2^{-S_p},
$$

a contradiction. ■