

# Unifying Leakage Classes: Simulatable Leakage and Pseudoentropy\*

Benjamin Fuller<sup>†</sup>      Ariel Hamlin<sup>‡</sup>

September 4, 2015

## Abstract

Leakage-resilient cryptography builds systems that withstand partial adversary knowledge of secret state. Ideally, leakage-resilient systems withstand current and future attacks; restoring confidence in the security of implemented cryptographic systems. Understanding the relation between classes of leakage functions is an important aspect.

In this work, we consider the memory leakage model, where the leakage class contains functions over the system’s entire secret state. Standard classes include functions with bounded output length, functions that retain (pseudo) entropy in the secret, and functions that leave the secret computationally unpredictable.

Standaert, Pereira, and Yu (Crypto, 2013) introduced a new class of leakage functions they call simulatable leakage. A leakage function is *simulatable* if a simulator can produce indistinguishable leakage without access to the true secret state. We extend their notion to general applications and consider two versions. For *weak* simulatability: the simulated leakage must be indistinguishable from the true leakage in the presence of public information. For *strong* simulatability, this requirement must also hold when the distinguisher has access to the true secret state. We show the following:

- Weakly simulatable functions retain computational unpredictability.
- Strongly simulatability functions retain pseudoentropy.
- There are bounded length functions that are not weakly simulatable.
- There are weakly simulatable functions that remove pseudoentropy.
- There are leakage functions that retain computational unpredictability are not weakly simulatable.

---

\*The Lincoln Laboratory portion of this work was sponsored by the Department of the Air Force under Air Force Contract #FA8721-05-C-0002. Opinions, interpretations, conclusions and recommendations are those of the author and are not necessarily endorsed by the United States Government.

<sup>†</sup>Boston University and MIT Lincoln Laboratory Email: [bfuller@cs.bu.edu](mailto:bfuller@cs.bu.edu).

<sup>‡</sup>MIT Lincoln Laboratory Email: [ariel.hamlin@ll.mit.edu](mailto:ariel.hamlin@ll.mit.edu)

# 1 Introduction

Cryptography relies on secret randomness, such as keys. It is crucial to properly model how an adversary can interact with and observe this secret state. As an example, when defining security of a block cipher, an adversary may ask for encryption of arbitrary plaintexts and see the corresponding ciphertext.. The secret key and randomness used by the algorithm are assumed to be hidden from the adversary.

Unfortunately, the adversary rarely uses cryptographic systems as black-boxes, exploiting side-channel information when possible. As many works have shown, side-channel attacks have been devastating to existing deployed cryptosystems [Koc96, KJJ99, BB05, BM06, OST06, TOS10, GST13].

Completely eliminating side-channel attacks seems hopeless. The cryptographic community began designing systems that remain secure in the presence of side-channel attacks. In the theory community, the work of Ishai, Sahai, and Wagner [ISW03] showed how to transform any circuit into one that withstood adversarial knowledge of some constant fraction of the wire values. The work of Micali and Reyzin [MR04] considered arbitrary leakage functions of bounded output length.

There are two crucial aspects to defining a leakage function: what the function computes on, and what type of computations the function can perform. We refer to these aspects as the leakage *model* and leakage *class* respectively.

**Leakage Models** There are two common models: *circuit leakage* assumes the leakage function operates on a particular circuit implementation of a computation. The leakage function is allowed (with restrictions) to compute on individual gates and wires in the circuit. *Memory leakage* allows the leakage function to leak on secret state and ignores the intermediate states of computation specific to the implementation of the algorithm.<sup>1</sup> Circuit leakage is used in conjunction with leakage classes restricted to local computations. Memory leakage is used in conjunction with functions that access all state simultaneously (but with some restriction on the output). Results in these models are not easily compared. We focus on the memory leakage model but briefly discuss simulatable leakage in the context of circuit leakage in Appendix A.

**Leakage Classes** We now describe common leakage classes in the memory leakage model.

- **Bounded Length** [MR04]. The leakage function is an arbitrary function  $\mathcal{L}$  of input, secret state and randomness. The only limitation on the function is a bounded output length. A natural broadening of this class is

---

<sup>1</sup>If the leakage class is sufficiently powerful, the particular implementation of an algorithm is irrelevant. The leakage function can recompute a given implementation of the functionality. This is the case for all leakage classes we consider in the memory leakage model.

the set of all functions that preserve min-entropy of the secret state.<sup>2</sup> We denote this class of leakage functions by **bLEN**.

- **Indistinguishable** [DP08] Bounded length leakage is easy to reason about because secret state has entropy conditioned on the output of the leakage function. Unfortunately, many leakage functions (such as the power trace of computation) are quite long. Furthermore, in many applications the secret key does not have information-theoretic entropy (for example, a Diffie-Hellman key conditioned on the public transcript). A leakage function  $\mathcal{L}$  is an *indistinguishable leakage function* if the secret state looks like it has entropy conditioned on  $\mathcal{L}$  (we use HILL entropy [HILL99]). Note this class contains bounded length functions. We denote this class of leakage functions by **Indist**. We also use a weaker notion of pseudoentropy called relaxed HILL entropy [Rey11] and denote the class of leakage functions that preserve relaxed HILL entropy as **rIndist**.
- **Hard-to-invert** [DKL09]. The indistinguishable leakage model is too restrictive for many applications. As an example, a symmetric cipher key is often uniquely determined conditioned on a few plaintext/ciphertext pairs. It is usually possible to verify a guess for the key and thus, it is not indistinguishable from any high entropy distribution. A minimum condition is that secret state is hard to guess given leakage. This is known as *hard-to-invert* leakage. We denote this class of functions as **hINV**.
- **Simulatable** [SPY13]. Standaert, Pereira, and Yu recently introduced simulatable leakage. Consider some private state  $K$  with some public information  $Y$  (such as a public key or plaintext/ciphertext pairs). A leakage function is simulatable if a simulator  $S$  can create a random variable  $S(Y)$  that is indistinguishable from  $\mathcal{L}(K)$ . Simulatable leakage is a combination of ideas from practice and theory. It allows simulators to be proposed for actual leakage functions. Then practitioners can try and distinguish simulator output from the true leakage. Indeed, the simulator proposed by Standaert et al. was subsequently broken [LMO<sup>+</sup>14]. The work of Standaert, Pereira, and Yu also shows how to construct a stream cipher that withstands simulatable leakage from a pseudorandom generator that withstands simulatable leakage.

Containments between the first three leakage classes are understood. (**bLEN**  $\subset$  **Indist**  $\subset$  **hINV**.) Simulatable leakage is a natural definition. Ideally, simulatable leakage would preserve security as an adversary could use the simulated leakage (and execute their attack with similar success probability) and therefore leakage would not harm application security. The goal of this work is to clarify this intuition.

We consider two versions of simulatable leakage: first where the simulated leakage must be consistent with only the public system state, and second where

---

<sup>2</sup>If the output length of the leakage function is significantly less than the entropy of the secret state, then bounded length leakage functions retain average min-entropy.

the simulated leakage must be consistent with both the public and private system state. We call these classes weakly simulatable (**wSIM**) and strongly simulatable (**sSIM**) respectively.

**Meaningfulness of Weakly Simulatable Leakage** Weakly simulatable leakage is not always meaningful. As example, consider an adversary trying to guess a private key  $K$  with no public information. The identity function is a weakly simulatable. A simulator for the leakage can sample a uniform random key  $K'$ . We call this situation *leak-and-resample*. To prevent this, we assume it is difficult to sample a key that is consistent with the public information (Definition 3.6). This is the case in many applications but not all. We call this setting—borrowing terminology from zero knowledge—*witness hiding*.

**Our contribution:** We connect the notion of simulatable leakage to standard leakage models. A graphical description of our results is in Figure 1. We show five results:

- Lemma 4.1: Strong simulatability implies relaxed HILL entropy. That is,  $\mathbf{sSIM} \subseteq \mathbf{rIndist}$ .
- Lemma 5.1: There are simulatable leakage functions that remove all pseudentropy from private state. That is,  $\mathbf{wSIM} \not\subseteq \mathbf{rIndist}$ .
- Lemma 5.6: There are bounded-length leakage functions that are not simulatable.  $\mathbf{bLEN} \not\subseteq \mathbf{wSIM}$ .
- Lemma 6.1: Simulatable leakage preserves unpredictability.  $\mathbf{wSIM} \subset \mathbf{hINV}$ .
- Lemma 6.3: There are hard-to-invert leakage functions that are not simulatable.  $\mathbf{hINV} \not\subseteq \mathbf{wSIM}$ .

**Discussion:** These results show that weakly simulatable leakage is properly contained in hard-to-invert leakage. This suggests it may be possible to build cryptographic primitives for weakly simulatable leakage that have eluded hard-to-invert leakage. Building crypto systems secure against hard-to-invert leakage been difficult [FHN<sup>+</sup>12], suggesting that simulatable leakage is a promising alternative.

This work places simulatable leakage in the context of other memory leakage classes. The complementary question is how simulatable leakage fits with previously considered leakage classes in the circuit model. We discuss definitional considerations for simulatable leakage in the circuit model in Appendix A. Providing results in the circuit model is more complicated as one must consider the implementation of a functionality. We leave this an open problem.

In this work we show that **sSIM** is contained in **rIndist**. It seems natural that simulatable leakage is related to indistinguishability (since it is an indistinguishability based definition). Settling the containment with **Indist** is an interesting question.

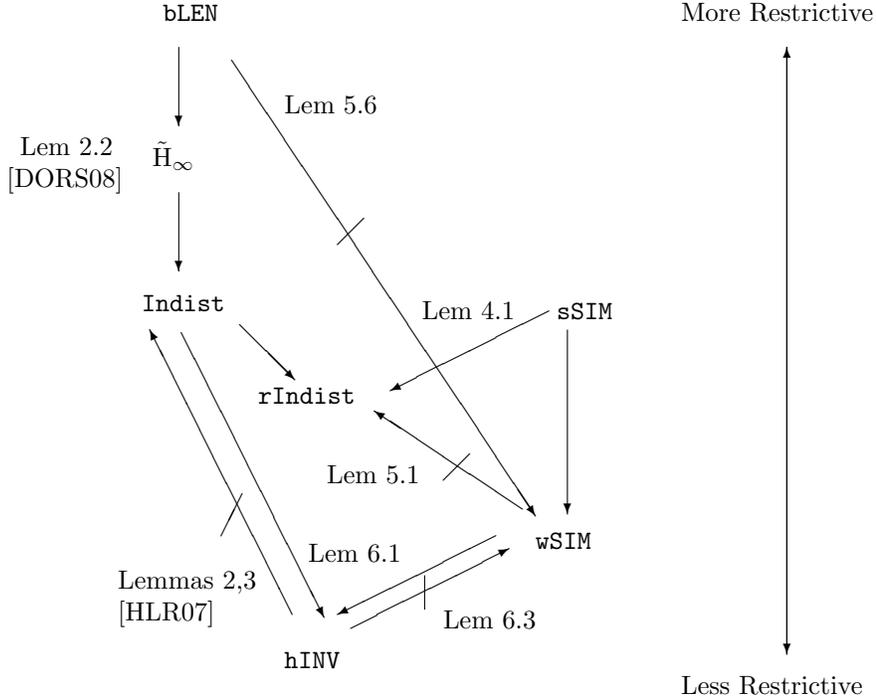


Figure 1: Containment between difference leakage notions. The relations between simulatable leakage are shown in this work. Arrows imply containment and arrows with slashes imply there is a function in one class not contained in the other class.

**Organization:** The remainder of the paper is organized as follows. We begin by covering preliminaries and definitions of memory leakage classes in Section 2. In Section 3, we define simulatable leakage and extend it to general applications (the definition of [SPY13] is specific to symmetric ciphers). In Section 4, we discuss strong simulatability and pseudoentropy. In Sections 5 and 6, we connect weakly simulatable leakage to indistinguishable and hard-to-invert leakage respectively.

## 2 Preliminaries

We usually use upper case letters for random variables and lower case letters for particular outcomes.  $U_n$  denotes the uniformly distributed random variable on  $\{0, 1\}^n$ . Unless otherwise noted logarithms are base 2.

**Entropy Notions** The *min-entropy* of  $X$  is  $H_\infty(X) = -\log(\max_x \Pr[X = x])$ . Let  $|W|$  be the size of the support of  $W$  that is  $|W| = |\{w \mid \Pr[W = w] > 0\}|$ .

**Definition 2.1.** [DORS08, Section 2.4] The average (conditional) min-entropy of  $X$  given  $Y$  is

$$\tilde{H}_\infty(X|Y) = -\log\left(\mathbb{E}_{y \in Y} \max_x \Pr[X = x|Y = y]\right).$$

**Distance Notions** The *statistical distance* between random variables  $X$  and  $Y$  with the same domain is  $\Delta(X, Y) = \frac{1}{2} \sum_x |\Pr[X = x] - \Pr[Y = x]|$ . For a distinguisher  $D$  (or a class of distinguishers  $\mathcal{D}$ ) we write the *computational distance* between  $X$  and  $Y$  as  $\delta^{\mathcal{D}}(X, Y) = |\mathbb{E}[D(X)] - \mathbb{E}[D(Y)]|$ . We denote by  $\mathcal{D}_{s_{sec}}$  the class of randomized circuits which output a single bit and have size at most  $s_{sec}$ .

We use a slightly nonstandard notion of a one-way function is hard on a particular input distribution (instead of the uniform distribution)

**Definition 2.2.** Let  $K$  be a distribution over space  $\mathcal{M}$  and let  $f : \mathcal{M} \rightarrow \{0, 1\}^*$ . We say that  $f$  is  $(s, \epsilon, K)$ -one-way if for all  $A$  of size at most  $s$ ,  $\Pr_{x \leftarrow K}[f(A(f(x))) = f(x)] \leq \epsilon$ .

## 2.1 Pseudoentropy

In this section, we present notions of pseudoentropy that are used to describe leakage classes. Pseudoentropy is the computational analogue of min-entropy. In general, a pseudoentropy notion describes how much entropy a random variable has to computationally bounded adversaries. The most common notions of pseudoentropy consider indistinguishability from a high entropy random variable and unpredictability.<sup>3</sup>

**Indistinguishability** We use the common notion of HILL entropy [HILL99] extended to the conditional setting [HLR07].

**Definition 2.3.** [HLR07] Let  $(K, Y)$  be a pair of random variables.  $K$  has relaxed HILL entropy at least  $k$  conditioned on  $Y$ , denoted  $H_{\epsilon_{ent}, s_{ent}}^{\text{HILL}}(K|Y) \geq k$  if for each  $y \in Y$  there exists distributions  $Z_y$  giving rise to a joint distribution  $(Z, Y)$ , such that  $\tilde{H}_\infty(Z|Y) \geq k$  and  $\delta^{\mathcal{D}_{s_{ent}}}(K, Y), (Z, Y) \leq \epsilon_{ent}$ .

One of the primary uses of HILL entropy is that applying a randomness extractor [NZ93] yields pseudorandom bits [BSW03, Lemma 4.2]. There are many notions of indistinguishability based pseudoentropy [BSW03, Sko14]. One significant drawback of conditional HILL entropy is that revealing one bit can significantly decrease HILL entropy [KPW13]. Relaxed HILL entropy allows replacement of the condition as well in the indistinguishability game.

**Definition 2.4.** [GW11, Rey11] Let  $(K, Y)$  be a pair of random variables.  $K$  has relaxed HILL entropy at least  $k$  conditioned on  $Y$ , denoted  $H_{\epsilon_{ent}, s_{ent}}^{\text{HILL-rlx}}(K|Y) \geq k$  if there exists a joint distribution  $(K', Y')$  such that  $\tilde{H}_\infty(K'|Y') \geq k$  and  $\delta^{\mathcal{D}_{s_{ent}}}(K, Y), (K', Y') \leq \epsilon_{ent}$ .

<sup>3</sup>There are also notions of pseudoentropy that consider compressibility of a random variable. We do not discuss this notion in this work. See [Yao82, BSW03, HLR07].

Relaxed HILL entropy is a weaker notion than HILL entropy (by restricting to the joint distributions  $(K', Y')$  where  $Y' = Y$ ).

**Proposition 2.5.**  $H_{\epsilon_{ent}, s_{ent}}^{\text{HILL-rlx}}(K|Y) \geq H_{\epsilon_{ent}, s_{ent}}^{\text{HILL}}(K|Y)$ .

However, it is still useful as applying a randomness extractor still yields a pseudorandom output [Ful15, Theorem 2.2.4]. Furthermore, relaxed HILL entropy obeys a chain rule unlike traditional HILL entropy [GW11, Rey11].

**Unpredictability** One can also consider the unpredictability of a random variable by computationally bounded adversaries. This is captured by the following definition [HLR07]:

**Definition 2.6.** Let  $(K, Y)$  be a pair of random variables. We say that  $K$  has unpredictability entropy at least  $k$  conditioned on  $Y$ , denoted by  $H_{\epsilon_{unp}, s_{unp}}^{\text{unp}}(K|Y) \geq k$ , if for all joint distributions  $(Z, Y)$  such that  $\delta^{\mathcal{D}_{s_{unp}}}((K, Y), (Z, Y)) \leq \epsilon_{unp}$ , and for all circuits  $\mathcal{I}$  of size  $s_{unp}$ ,

$$\Pr[\mathcal{I}(Y) = Z] \leq 2^{-k}.$$

HILL entropy is at least as large as unpredictability entropy.

**Proposition 2.7.** [HLR07, Lemma 8]  $H_{\epsilon, s}^{\text{HILL}}(K|Y) \geq H_{\epsilon, s}^{\text{unp}}(K|Y)$ .

The work of Hsiao, Lu, and Reyzin shows they can be separated by an arbitrary polynomial in the conditional case [HLR07, Lemmas 2 and 3]. However, it is possible extract from unpredictability entropy using a randomness extractor with a reconstruction procedure [HLR07, Lemma 6].

In our results, we use of the fact that HILL and unpredictability entropy are unchanged if a polynomial size circuit is applied to the condition. In the information theoretic setting this is known as the data processing inequality.

**Lemma 2.8.** Let  $S$  be a circuit of size  $s_{sim}$ . Then

$$H_{\epsilon_{ent}, s_{ent} - s_{sim}}^{\text{HILL}}(K|S(Y)) \geq H_{\epsilon_{ent}, s_{ent}}^{\text{HILL}}(K|Y).$$

*Proof.* Let  $H_{\epsilon_{ent}, s_{ent}}^{\text{HILL}}(K|Y) = k$ . Suppose for the sake of contradiction that for all joint distributions  $Z, S(Y)$  such that  $\tilde{H}_{\infty}(Z|S(Y)) \geq k$  there exists  $D \in \mathcal{D}_{s_{ent} - s_{sim}}$  such that

$$D((Z, S(Y)), (K, S(Y))) > \epsilon_{ent}.$$

Let  $Z', Y$  be a distribution such that  $\tilde{H}_{\infty}(Z'|Y) \geq k$ . By the information-theoretic data-processing inequality,  $\tilde{H}_{\infty}(Z'|S(Y)) \geq k$ . Thus, there exists a  $D \in \mathcal{D}_{s_{ent} - s_{sim}}$  such that  $D((Z, S(Y)), (K, S(Y))) > \epsilon_{ent}$ . Fix one such  $D$ . Consider the distinguisher  $D'(z, y) = D(z, S(y))$  (of size at most  $s_{ent}$ ). Then  $D'((K, Y), (Z', Y)) \geq D((K, S(Y)), (Z', S(Y))) \geq \epsilon_{ent}$ . This is a contradiction and completes the proof.  $\square$

This fact also holds for relaxed HILL entropy and unpredictability entropy.

**Lemma 2.9.** *Let  $S$  be a circuit of size  $s_{sim}$ . Then*

$$H_{\epsilon_{ent}, s_{ent}-s_{sim}}^{\text{HILL-rlx}}(K|S(Y)) \geq H_{\epsilon_{ent}, s_{ent}}^{\text{HILL-rlx}}(K|Y).$$

*Proof.* Let  $H_{\epsilon_{ent}, s_{ent}}^{\text{HILL-rlx}}(K|Y) = k$ . Suppose for the sake of contradiction that for all join distributions  $K', Y'$  such that  $\tilde{H}_{\infty}(K'|Y') \geq k$  there exists  $D \in \mathcal{D}_{s_{ent}-s_{sim}}$  such that

$$D((K, S(Y)), (K', Y')) > \epsilon_{ent}.$$

Let  $X', Z'$  be a distribution such that  $\tilde{H}_{\infty}(X'|Z') \geq k$ . By the information-theoretic data-processing inequality,  $\tilde{H}_{\infty}(X'|S(Z')) \geq k$ . Thus, there exists a  $D \in \mathcal{D}_{s_{ent}-s_{sim}}$  such that

$$D((K, S(Y)), (X', S(Z'))) > \epsilon_{ent}.$$

Fix one such  $D$ . Consider the distinguisher  $D'(k, y) = D(k, S(y))$  (of size at most  $s_{ent}$ ). Then

$$D'((K, Y), (X', Z')) \geq D((K, S(Y)), (X', S(Z'))) \geq \epsilon_{ent}.$$

This is a contradiction and completes the proof.  $\square$

**Lemma 2.10.** *Let  $S$  be a circuit of size  $s_{sim}$ . Then*

$$H_{\epsilon_{ent}, s_{ent}-s_{sim}}^{\text{unp}}(K|S(Y)) \geq H_{\epsilon_{ent}, s_{ent}}^{\text{unp}}(K|Y).$$

*Proof.* Let  $H_{\epsilon_{ent}, s_{ent}}^{\text{unp}}(K|Y) = k$ . For the sake of arriving at a contradiction, assume that for all joint distributions  $Z, Y$  be a joint distribution such that  $\delta^{\mathcal{D}_{s_{unp}-s_{sim}}}((K, S(Y)), (Z, S(Y))) \leq \epsilon_{ent}$  there exists  $\mathcal{I}$  of size at most  $s_{unp}-s_{sim}$  such that  $\Pr[\mathcal{I}(S(Y)) = Z] > 2^{-k}$ .

Consider some distribution  $Z, Y$  such that  $\delta^{\mathcal{D}_{s_{unp}}}((K, Y), (Z, Y)) \leq \epsilon_{ent}$ . Then also  $\delta^{\mathcal{D}_{s_{unp}-s_{sim}}}((K, S(Y)), (Z, S(Y))) \leq \epsilon_{ent}$  (by the argument in Lemma 2.9). Thus, there exists an  $\mathcal{I}$  of size at most  $s_{unp}-s_{sim}$  such that  $\Pr[\mathcal{I}(S(Y)) = Z] > 2^{-k}$ . Consider the following inverter  $\mathcal{I}'$  (of size at most  $s_{unp}$ ):  $\mathcal{I}'(z) = \mathcal{I}(S(z))$ . Then

$$\Pr[\mathcal{I}'(Y) = Z] \Pr[\mathcal{I}(S(Y)) = Z] > 2^{-k}.$$

This is a contradiction and completes the proof.  $\square$

## 2.2 Leakage Models

In this section we focus on memory leakage models – the models in which our results focus. We briefly review circuit leakage models in Appendix A.

**Bounded Leakage:** This leakage class allows an arbitrary  $\mathcal{L}$  with limited output length [DP08].

**Definition 2.11.** *Let  $K$  be a discrete random variable over space  $\chi_1$ . The randomized map  $\mathcal{L} : \chi_1 \rightarrow \{0, 1\}^*$  is an  $\ell$ -bounded leakage function if for  $\mathcal{L}(x)$  takes at most  $2^\ell$  values for any choice  $x \in \chi_1$  and any choice of random coins of  $\mathcal{L}$ .*

For convenience, we refer to this class of leakage as **bLEN**. Bounded leakage is a natural definition. If a random variables starts with min-entropy  $k$ , we know that after  $\ell$  bits of leakage it has remaining min-entropy  $k - \ell$  [DORS08, Lemma 2.2]. That is, if  $H_\infty(K) \geq k$ , then  $\tilde{H}_\infty(K|\mathcal{L}(K)) \geq k - \ell$ . Unfortunately, bounded length leakage is not representative of reality. Many side channels take values in a universe larger than the key itself.

**Indistinguishable Leakage** In many applications, the secret state has no true information conditioned on the public state of the algorithm. For example, the secret key of a symmetric cipher has little entropy after a few plaintext/ciphertext pairs. However, it often has pseudoentropy. Dziembowski and Pietrzak construct a pseudorandom generator secure against this type of leakage [DP08]. Indistinguishability leakage retains high entropy (we refer to this class of functions as **Indist**):

**Definition 2.12.** *Let  $K$  be a random variable and let  $\mathcal{L}$  be a randomized map.  $\mathcal{L}$  is a  $(k, \epsilon_{ent}, s_{ent})$ -indistinguishable leakage function if  $H_{\epsilon_{ent}, s_{ent}}^{\text{HILL}}(K|\mathcal{L}(K)) \geq k$ .*

We refer to leakage functions that retain relaxed HILL entropy as **rIndist**.

**Hard to invert leakage** For a scheme with secret key  $K$ , the minimal notion of security is that an adversary should not be able to predict the value of  $K$ . This is model is known as the auxiliary input [DKL09] or hard-to-invert leakage [FHN<sup>+</sup>12]. We refer to this class of functions as **hINV**:

**Definition 2.13.** *Let  $K$  be a random variable over space  $\chi_1$ . The randomized map  $\mathcal{L}$  is a  $(k, \epsilon_{unp}, s_{unp})$ -hard-to-invert leakage if  $H_{\epsilon_{unp}, s_{unp}}^{\text{unp}}(K|\mathcal{L}(K)) \geq k$ .*

We make no condition in the above definition about  $K$  unconditionally. For  $K$  to be unpredictable with  $\mathcal{L}(K)$  it must have unconditional unpredictability at least  $k$ .<sup>4</sup> Hard-to-invert leakage seems like the weakest leakage class for which applications can retain security.

### 3 Simulatable Leakage

Standaert, Pereira, and Yu [SPY13] introduce a new leakage class designed to be achievable and verifiable. Simulatable leakage is leakage that can be

<sup>4</sup>In the unconditional setting, there is a polynomial time circuit that predicts  $K$  with probability  $2^{-H_\infty(K)}$ . That is,  $H_\infty(K) = H_{\epsilon_{unp}, s_{unp}}^{\text{unp}}(K)$ .

simulated without access to the true secret state. We first present the definition of Standaert, Pereira, and Yu [SPY13]. This definition is specific to the setting of a block cipher (denoted  $BC$ ) in the presence of leakage function  $\mathcal{L}$ . See [SPY13] for more information on block ciphers and discussion.

Game $sim(q, D, BC, \mathcal{L}, S, b)$ .		
The challenger selects two random keys $k$ and $k^*$ in $\{0, 1\}^n$ . The output of the game is a bit $b'$ computed by $D$ based on the challenger responses to a total of at most $q$ adversarial queries of the following type:		
Query	Response if $b = 0$	Response if $b = 1$
$Enc(x)$	$BC_k(x), \mathcal{L}(k, x)$	$BC_{k^*}(x), S(k^*, x, BC_{k^*}(x))$
and one query of the following type:		
Query	Response if $b = 0$	Response if $b = 1$
$Gen(z, x)$	$S(z, x, k)$	$S(z, x, k^*)$

**Definition 3.1.** [SPY13, Definition 1] *A block cipher  $BC$  with leakage function  $\mathcal{L}$  has  $(\epsilon, s_{sim}, s_{sec})$   $q$ -simulatable leakages if there is a simulator  $S$ , of size  $s_{sim}$ , for every  $D$ , of size  $s_{sec}$ , we have:*

$$\delta(sim(q, D, BC, \mathcal{L}, S, 1), sim(q, D, BC, \mathcal{L}, S, 0)) \leq \epsilon.$$

### 3.1 Extending Simulatable leakage to general applications

Definition 3.1 is specialized to the setting of symmetric-key cryptography. In particular, the second type of query exists because the authors argue that symmetric keys are often derived from sources that themselves have leakage. It is not clear how to generalize this type of query to arbitrary leakage settings. In addition, providing a single key to  $\mathcal{S}$  as consistent state is limiting, it is not clear why the simulator should not be allowed to keep state between leakage queries. Furthermore, the fact that leakage is provided with each output of the block cipher is not a necessary requirement. There may multiple leakage queries for each block cipher output or vice versa. Furthermore, the distinguisher does not have any access to  $k$  when trying to decide if the leakage is legitimate. In different applications, the distinguisher may have partial access to the secret state  $k$ . We present two definitions modeling the two extremes, one where the distinguisher has full access to the secret and one the secret is completely hidden. Our definitions consider two random variables  $K$  and  $Y$  that represent the private and public state of the cryptosystem (but we do not include this distinction in the definitions).

**Definition 3.2.** *Let  $(K, Y)$  be a pair of random variables over  $\chi_1 \times \chi_2$ . The randomized map  $\mathcal{L}$  is an  $(\epsilon, s_{sim}, s_{sec})$ -weakly simulatable leakage function if there exists a simulator  $S$  of size at most  $s_{sim}$  such that*

$$\delta^{\mathcal{D}_{s_{sec}}}((Y, \mathcal{L}(K)), (Y, S(Y))) \leq \epsilon.$$

**Definition 3.3.** Let  $(K, Y)$  be a pair of random variables over  $\chi_1 \times \chi_2$ . The randomized map  $\mathcal{L}$  is an  $(\epsilon, s_{sim}, s_{sec})$ -strongly simulatable leakage function if there exists a simulator  $S$  of size at most  $s_{sim}$  such that

$$\delta^{\mathcal{D}_{s_{sec}}}((K, Y, \mathcal{L}(K)), (K, Y, S(Y))) \leq \epsilon.$$

We use  $\mathfrak{wSIM}$  and  $\mathfrak{sSIM}$  as shorthand for weakly and strongly simulatable classes respectively.

**Proposition 3.4.** If  $\mathcal{L}$  is  $(\epsilon, s_{sim}, s_{sec})$ -strongly simulatable leakage function, then  $\mathcal{L}$  is a  $(\epsilon, s_{sim}, s_{sec})$ -weakly simulatable. That is,  $\mathfrak{wSIM} \subseteq \mathfrak{sSIM}$ .

**Notes:** These definitions do not model secret key updates. We assume a single leakage query. Alternatively, we can think of an adversary that prepares all of their multiple leakage queries simultaneously. This is slightly weaker than the definition of Standaert et al.

We also assume that  $Y$  incorporates all public values of the scheme. This may include a public-key, ciphertexts, signatures, etc. In the work of Standaert et al., this is assumed to be the input and output of the block cipher with the true key.

### 3.1.1 Meaningfulness of weakly simulatable leakage

Some restriction on  $Y$  is necessary to make weakly simulatable leakage meaningful. If  $Y$  is empty, the leakage  $\mathcal{L}(K) = K$  is simulatable by sampling a fresh secret key  $K'$ . However, there is no security remaining in the system. In particular, in this setting, there is no min-entropy, HILL entropy, or unpredictability entropy remaining in the key. Indeed, when there is no public state  $Y$ , any polynomial time function is simulatable (by sampling a fresh  $k \leftarrow K$  and outputting  $f(k)$ ). This gives us the following proposition:

**Proposition 3.5.** Let  $K$  be a random variable over  $\chi_1$  samplable by procedure **Sample** of size  $s_{sam}$  and let  $Y$  be empty. Let  $f : \chi_1 \rightarrow \{0, 1\}^*$  be a function computable by a circuit of size  $|f|$ . Then  $f$  is a  $(0, s_{sim}, \infty)$ -weakly simulatable leakage if  $s_{sim} \geq |f| + s_{sam}$ .

It is not just empty  $Y$  that presents a problem to weakly simulatable leakage. It may be possible to leak the entire secret even when it is information-theoretically determined by the condition  $Y$ . For example, if the public state is a Diffie-Hellman exchange  $g^a, g^b$ , then the key  $g^{ab}$  can be leaked (since a fresh  $g^c$  is indistinguishable).

To prevent these *leak-and-resample* simulators, we assume it is hard to find  $k$  values consistent with the public information. We use the notation of witness hiding from zero-knowledge.

**Definition 3.6.** Let  $K, Y$  be a joint random variable and let  $R$  be a relation (computable by a circuit of size  $s_{rel}$ ) where  $\Pr[R(K, Y) = 1] = 1$ . The public state  $Y$  is  $(s_{rel}, s_{inv}, \epsilon_{rel})$ -witness hiding if for all  $\mathcal{I}$  of size at most  $s_{inv}$ ,  $\Pr[R(\mathcal{I}(Y), Y) = 1] \leq \epsilon_{rel}$ .

**Note:** If it is hard to find keys that are consistent with plaintext/ciphertext pairs, then the definition of Standaert et al. also has a witness hiding condition.

When discussing weakly simulatable leakage, we consider public information that is witness hiding of the secret state  $K$ . Witness hiding implies unpredictability of  $H_{0,s_{inv}}^{\text{unp}}(K|Y) \geq -\log(\epsilon_{rel})$ . We first discuss strongly simulatable leakage and then weakly simulatable leakage. We discuss how to apply simulatable leakage to the circuit leakage model in Appendix A.

## 4 Strongly Simulatable Leakage

In this section, we show that all strongly simulatable leakages preserve indistinguishability. We use the relaxed notion of HILL entropy. We show that  $\text{sSIM} \subseteq \text{rIndist}$ . In the next section, we show that  $\text{bLEN} \not\subseteq \text{wSIM}$  which implies that  $\text{bLEN}$  (and thus  $\text{rIndist}$ ) are not contained in  $\text{sSIM}$ .

**Lemma 4.1** ( $\text{sSIM} \Rightarrow \text{rIndist}$ ). *Let  $K$  be a distribution over  $\chi$  and let  $Y$  be some public information. Let  $H_{\epsilon_{ent},s_{ent}}^{\text{HILL-r1x}}(K|Y) \geq k$  and let  $\mathcal{L}$  be a  $(\epsilon_{sim}, s_{sim}, s_{sec})$ -strongly simulatable leakage function. Then  $H_{\epsilon',s'}^{\text{HILL-r1x}}(K|Y, \mathcal{L}(K)) \geq k$  for  $\epsilon' = \epsilon_{ent} + \epsilon_{sim}, s' = \min\{s_{sec}, s_{ent} - s_{sim}\}$ .*

*Proof.* Fix  $\mathcal{L}$  and let  $S$  be a simulator of size at most  $s_{ent}$ . Define the circuit  $S'$  that on input  $y$  outputs  $y, S(y)$  and note that  $S'$  is of size  $s_{sim}$ . Then by Lemma 2.9 the simulator does not decrease relaxed HILL entropy,

$$H_{\epsilon_{ent},s_{ent}-s_{sim}}^{\text{HILL-r1x}}(K|Y, S(Y)) = H_{\epsilon_{ent},s_{ent}-s_{sim}}^{\text{HILL-r1x}}(K|S'(Y)) \geq k.$$

Thus, there exists some  $K', Y', Z'$  where  $\tilde{H}_{\infty}(K'|Y', Z') \geq k$  such that

$$\delta^{\mathcal{D}^{s_{ent}-s_{sim}}}((K, Y, S(Y)), (K', Y', Z')) \leq \epsilon_{ent}.$$

By simulatability, we have that

$$\delta^{\mathcal{D}^{s_{sec}}}((K, Y, \mathcal{L}(K)), (K, Y, S(Y))) \leq \epsilon_{sim}.$$

Finally, by the triangle inequality,

$$\delta^{\mathcal{D}^{\min\{s_{sec}, s_{ent}-s_{sim}\}}}((K, Y, \mathcal{L}(K)), (K', Y', Z')) \leq \epsilon_{sim} + \epsilon_{ent}.$$

□

## 5 wSIM and rIndist

In the previous section, we showed that strong simulatability of a leakage function implied relaxed HILL entropy. However, this does not carry over to the setting of weak simulatability. In this section we show that simulatable leakage is incomparable with indistinguishable leakage functions. We show a

bounded leakage function that is not simulatable ( $\mathbf{bLEN} \not\subseteq \mathbf{wSIM}$ ) and a simulatable leakage function that removes all relaxed pseudoentropy from the secret ( $\mathbf{rIndist} \not\subseteq \mathbf{wSIM}$ ). Since bounded length leakage functions are contained in  $\mathbf{Indist}$  (and  $\mathbf{rIndist}$ ) this also shows that  $\mathbf{rIndist} \not\subseteq \mathbf{wSIM}$ . In this section, we assume that the public information is witness hiding of the secret state (Definition 3.6).

## 5.1 $\mathbf{wSIM} \not\subseteq \mathbf{rIndist}$

There are simulatable leakage functions that remove all  $\mathbf{HILL-r1x}$ .

**Lemma 5.1** ( $\mathbf{wSIM} \not\subseteq \mathbf{rIndist}$ ). *Let  $K = (K_1, K_2)$  where  $K_1 \in \{0, 1\}^{\ell_1}$  and  $K_2 \in \{0, 1\}^{\ell_2}$  be uniformly distributed. Let  $f$  be an  $(\epsilon_{owp}, s_{owp})$ -injective one-way function from  $\{0, 1\}^{\ell_1} \rightarrow \{0, 1\}^{\ell_3}$  computable in size  $|f|$ . Let  $Y = f(K_1)$ . Then  $H_\infty(K|Y) = \ell_2$ . The function  $\mathcal{L}(K) = K_2$  is  $(0, \ell_2, \infty)$ -weakly simulatable and  $H_{\epsilon, s_{sec}}^{\mathbf{HILL-r1x}}(K|Y, \mathcal{L}(K)) \leq -\log(1 - \epsilon)$  if  $s_{sec} \geq s_{owf} + \ell_1 + \ell_2$ .*

*Proof.* We first show that  $Y, \mathcal{L}(K)$  removes all  $\mathbf{HILL-r1x}$  from  $K$ .

**Lemma 5.2.** *Let  $K$  be a random variable over  $\{0, 1\}^\ell, \{0, 1\}^{\ell_2}$  and let  $f : \{0, 1\}^{\ell_1} \rightarrow \{0, 1\}^{\ell_2}$  be an injective function computable by a circuit of size  $|f|$ . Then*

$$H_{\epsilon, |f| + \ell_1 + \ell_2}^{\mathbf{HILL-r1x}}(K|f(K)) \leq -\log(1 - \epsilon).$$

*Proof.* Let  $K', Z'$  be a distribution

$$\delta^{\mathcal{D}^{s_{sec}}}((K, f(K)), (K', Z')) \leq \epsilon.$$

Consider the distinguisher

$$D(k, z) = 1 \text{ if and only if } f(k) = z.$$

Clearly  $\mathbb{E}[D(K, f(K))] = 1$ . By indistinguishability,  $\mathbb{E}[D(K', Z')] \geq 1 - \epsilon$ . This means that

$$\mathbb{E}_{z \leftarrow Z'} D(K'|Z' = z, z) \geq 1 - \epsilon.$$

For all  $z$  there is a unique  $k \in \{0, 1\}^{\ell_1}$  such that  $D(k, z) = 1$ , denote this value by  $k_z$ . This means that  $\mathbb{E}_{z \leftarrow Z'} \Pr[K' = k_z | Z' = z] \geq 1 - \epsilon$ . We then have the following:

$$\mathbb{E}_{z \leftarrow Z'} \max_k \Pr[K' = k | Z' = z] \geq \mathbb{E}_{z \leftarrow Z'} \Pr[K = k_z | Z' = z] \geq 1 - \epsilon.$$

Taking the negative logarithm of each side yields that  $\tilde{H}_\infty(K'|Z') \leq -\log(1 - \epsilon)$ . This completes the proof of Lemma 5.2.  $\square$

We now return to the proof of Lemma 5.1. Lemma 5.2 implies that

$$H_{\epsilon, s_{sec}}^{\mathbf{HILL-r1x}}(K|Y, \mathcal{L}(K)) \leq -\log(1 - \epsilon).$$

$Y$  is a  $(|f|, s_{owp}, \epsilon_{owp})$ , witness hiding relation of  $K$  where the relation is  $R(k_1, k_2, y) = (f(k_1) \stackrel{?}{=} y)$ . The simulator  $S$  for  $\mathcal{L}$  computes a uniform sample from  $\{0, 1\}^{\ell_2}$ . This is identically distributed to  $\mathcal{L}(K)$  and takes  $\ell_2$  size to compute. Since there is a unique  $k_1$  for each  $y$ ,  $\tilde{H}_\infty(K|Y) = \tilde{H}_\infty(K_2|Y) = H_\infty(K_2)$ . This completes the proof of Lemma 5.1.  $\square$

## 5.2 bLEN $\not\subseteq$ wSIM

We now show a leakage function of bounded length that cannot be simulated.<sup>5</sup> We will use a secure signature scheme and leak a valid signature. This leakage function has been used previously to demonstrate the difficulty of constructing leakage resilient signature schemes [FHN<sup>+</sup>12]. We need a signature scheme that is hard to forge and a signature does not determine the secret key completely. We begin by describing the EU-RMA notion of signatures from Goldwasser, Micali, and Rivest [GMR88].

**Definition 5.3** (EU-RMA). *A signature scheme  $\Sigma = (\text{Gen}, \text{Sig}, \text{Ver})$  is  $(q, s_{sec}, \epsilon)$ -existential unforgeable against random message attacks if for all circuits  $A$  of size  $s_{sec}$  the following holds:*

$$\Pr_{(pk, sk) \leftarrow \text{Gen}(\cdot)} [m_1, \dots, m_q \leftarrow \mathcal{M} \wedge \sigma_i \leftarrow \text{Sig}(m_i, sk) \wedge (m^*, \sigma^*) \leftarrow A(m_1, \dots, m_q, \sigma_1, \dots, \sigma_q, pk) \wedge m^* \neq m_i \wedge \text{Ver}(pk, m^*, \sigma^*) = 1] < \epsilon$$

Under this definition a signature must not be simulatable. To ensure that the secret key still has high entropy we need a signature scheme where multiple private keys exist for each public key. We use a scheme where it is hard to find a candidate private key for each public key (making the public key witness hiding). We use Lamport's one-time secure signature scheme [Lam79].<sup>6</sup>

**Construction 5.4.** *Let  $f$  be a  $(\epsilon_{owf}, s_{owf})$ -one-way function mapping  $\{0, 1\}^{k^c} \rightarrow \{0, 1\}^k$  for  $c > 1$ :*

**Key Generation:** *Choose random  $x_{i,0}, x_{i,1} \leftarrow \{0, 1\}^{k^c}$  for  $i = 1, \dots, \ell$ . Compute  $y_{i,b} \leftarrow f(x_{i,b})$  for  $i \in \{1, \dots, \ell\}$  and  $b \in \{0, 1\}$ . The public key is  $pk = \{y_{i,b}\}$  and the secret key is  $\{x_{i,b}\}$ .*

**Signing:** *The signature on a  $k$ -bit message  $m = m_1, \dots, m_k$  consists of the  $k$  values  $x_{1,m_1}, \dots, x_{k,m_k}$ .*

**Verification:** *Given  $x_1, \dots, x_k$  and  $m = m_1, \dots, m_k$  and  $pk = (s, \{y_{i,b}\})$ , output 1 iff  $y_{i,m_i} \stackrel{?}{=} f(x_i)$  for all  $i$ .*

<sup>5</sup>This also shows that  $\text{Indist} \not\subseteq \text{wSIM}$  and  $\text{rIndist} \not\subseteq \text{wSIM}$ .

<sup>6</sup>This scheme was used in the setting of leakage-resilient cryptography by Katz and Vaikuntanathan [KV09]. They extend Lamport's scheme making the function collision resistant and using error correcting codes. Lamport's original scheme suffices for our purposes.

**Lemma 5.5.** *Construction 5.4 is a  $(1, \epsilon_{owf}, s_{owf})$ -secure signature scheme. Furthermore  $\tilde{H}_\infty(SK|PK) \geq 2\ell k^c - 2\ell k$ . Furthermore for any message  $m$ ,*

$$\tilde{H}_\infty(SK|PK, \text{Sig}_{PK}(m)) \geq \ell k^c - 2\ell k.$$

*Proof.* We omit the proof that the scheme is secure (and that  $Y = PK$  is witness hiding of  $X$ ). We have the following for the entropy calculations by [DORS08, Lemma 2.2],  $\tilde{H}_\infty(SK|PK) \geq 2\ell k^c - 2\ell k$ . Similarly, for any  $m$   $\tilde{H}_\infty(SK|PK, \text{Sig}_{PK}(m)) \geq 2\ell k^c - 2\ell k - \ell k^c \geq \ell k^c - 2\ell k$ .  $\square$

**Lemma 5.6.** *Let  $(\text{Gen}, \text{Sig}, \text{Ver})$  be as above for some  $c > 1$ , let  $K = SK, Y = PK$ . Then for any message  $m$  the function  $\mathcal{L}(K) = \text{Sig}_{SK}(m)$  is not simulatable by any  $S$  of size  $s_{sec} \leq s_{owf}$  with  $\epsilon \leq \epsilon_{owf}$ . Furthermore,  $\tilde{H}_\infty(SK|PK, \mathcal{L}(K)) \geq \ell k^c - 2\ell k$ .*

*Proof.* The lack of a simulator follows from the one-time security of the signature scheme. The remaining entropy follows from Lemma 5.5.  $\square$

## 6 wSIM $\not\subseteq$ hINV

In the previous section, we showed that weakly simulatable leakage and indistinguishable leakage are incomparable. In this section, we turn to hard-to-invert leakage. We show that weakly simulatable leakage preserves unpredictability but there are leakage functions that preserve unpredictability that are not simulatable. Our results assume  $Y$  is witness hiding.

### 6.1 wSIM $\subseteq$ hINV

The ability to predict  $K$  given both  $Y$  and  $\mathcal{L}(K)$  is not significantly different than the ability to predict the witness given just  $Y$ .

**Lemma 6.1.** *Let  $K, Y$  be a pair of random variables and let  $R$  be a  $(s_{rel}, s_{inv}, \epsilon_{rel})$ -witness hiding relation on  $K, Y$ . If  $\mathcal{L}$  be a  $(\epsilon_{sim}, s_{sim}, s_{sec})$ -weakly simulatable leakage for  $(K, Y)$ . Then  $H_{0, s'_{inv}}^{\text{unp}}(K|Y, \mathcal{L}(K)) \geq -\log(\epsilon_{rel} + \epsilon_{sim})$  for  $s'_{inv} = \min\{s_{sec} - s_{rel}, s_{inv} - s_{rel}\}$ .*

*Proof.* Let  $S$  be a simulator of size  $s_{sim}$  for  $\mathcal{L}$ . Suppose there exists an inverter  $\mathcal{I}$  of size  $s'_{inv}$  such that  $\Pr[\mathcal{I}(Y, \mathcal{L}(K)) = K] > \epsilon_{rel} + \epsilon_{sim}$ . To arrive at a contradiction it suffices to show there exists an inverter  $\mathcal{I}'(Y)$  of size  $s'_{inv} + s_{sim} \leq s_{inv}$  and succeeds with probability  $> \epsilon_{rel}$ . Define  $\mathcal{I}'(y) = \mathcal{I}(y, S(y))$ .

**Claim 6.2.**  $\Pr[R(\mathcal{I}(Y, S(Y)), Y) = 1] \geq \Pr[R(\mathcal{I}(Y, \mathcal{L}(K)), K) = 1] - \epsilon_{sim} \geq \epsilon_{rel}$ .

*Proof.* Recall that  $\delta^{\mathcal{D}_{s_{sec}}}((Y, \mathcal{L}(K)), (Y, S(Y))) \leq \epsilon_{sim}$ . Suppose for contradiction that

$$\Pr[R(\mathcal{I}'(Y, S(Y)), K) = 1] < \Pr[R(\mathcal{I}'(Y, \mathcal{L}(K)), K) = 1] - \epsilon_{sim}.$$

We present a distinguisher  $D$  of size  $s'_{inv} + s_{rel} \leq s_{sec}$ :

- On input  $y, z$ .
- Run  $x \leftarrow \mathcal{I}'(y, z)$ .
- Output 1 if and only if  $R(x, y) = 1$ .

Then

$$\begin{aligned} & \Pr[D(Y, \mathcal{L}(K)) = 1] - \Pr[D(Y, S(Y)) = 1] \\ &= \Pr[R(\mathcal{I}(Y, \mathcal{L}(K)), K) = 1] - \Pr[R(\mathcal{I}(Y, S(Y)), K) = 1] > \epsilon_{sim}. \end{aligned}$$

This is a contradiction. This completes the proof of the claim and the proof of the lemma.  $\square$

$\square$

## 6.2 hINV $\not\subseteq$ wSIM

In the previous section, we saw that simulatable leakage preserves unpredictability. In this section, we show this containment is tight.

**Lemma 6.3.** *Let  $f_1 : \{0, 1\}^{\ell_1} \rightarrow \{0, 1\}^{\ell_2}$  be  $(s_1, \epsilon_1, U_{\ell_1})$ -one way and let  $f_2 : \{0, 1\}^{\ell_2} \rightarrow \{0, 1\}^{\ell_3}$  be  $(s_2, \epsilon_2, f(U_{\ell_1}))$ -one way. Then for  $K = U_{\ell_1}$ ,  $Y = f_2(f_1(K))$ ,  $\mathcal{L}(K) = f_1(K)$  the following hold:*

1.  $H_{0, s_1 - |f_2|}^{\text{unp}}(K|Y, \mathcal{L}(K)) \geq -\log(\epsilon_{owf, 1})$ .
2.  $\mathcal{L}$  is not  $(|f_2|, s_2, 1 - \epsilon_2)$ -weakly simulatable.

*Proof.* We prove each statement in turn. Suppose Statement 1 is not true, that is, there exists an inverter  $\mathcal{I}'$  of size  $s_1 - |f_1|$  that inverts  $f_2 \circ f_1|_{f_1}$ . Then  $\mathcal{I}(y) = \mathcal{I}'(y, f_2(y))$  is an inverter for  $f_1$ .

Now suppose that Statement 2 is not true. Then there exists a simulator  $S$  of size  $s_2$  that simulates  $f_1(K)$ . That is,  $\delta^{\mathcal{D}_{sec}}((Y, \mathcal{L}(K)), (Y, S(Y))) < 1 - \epsilon_2$ . Consider the following distinguisher  $D$  (of size  $|f_2|$ ):

- Input  $y, z$ .
- Output 1 if and only if  $y = f_2(z)$ .

Clearly,  $\mathbb{E}[D(Y, \mathcal{L}(K))] = 1$ . Thus, by indistinguishability,  $\mathbb{E}[D(Y, S(K))] \geq 1 - (1 - \epsilon_2) \geq \epsilon_2$ , this is a contradiction.  $\square$

## Acknowledgements

The authors wish to thank Olivier Pereira, Krzysztof Pietrzak, Leonid Reyzin, François-Xavier Standaert, and Sophia Yakoubov for their helpful and insightful discussion. In addition, the authors wish to thank the anonymous reviewers for their constructive and astute comments. The work of Benjamin Fuller is sponsored in part by US NSF grants 1012910 and 1012798.

## References

- [BB05] David Brumley and Dan Boneh. Remote timing attacks are practical. *Computer Networks*, 48(5):701–716, 2005.
- [BM06] Joseph Bonneau and Ilya Mironov. Cache-collision timing attacks against AES. In *Cryptographic Hardware and Embedded Systems-CHES 2006*, pages 201–215. Springer, 2006.
- [BSW03] Boaz Barak, Ronen Shaltiel, and Avi Wigderson. Computational analogues of entropy. In *11th International Conference on Random Structures and Algorithms*, pages 200–215, 2003.
- [CJRR99] Suresh Chari, Charanjit Jutla, Josyula Rao, and Pankaj Rohatgi. Towards sound approaches to counteract power-analysis attacks. In *Advances in Cryptology-CRYPTO 1999*, pages 398–412. Springer, 1999.
- [DDF14] Alexandre Duc, Stefan Dziembowski, and Sebastian Faust. Unifying leakage models: From probing attacks to noisy leakage. In *Advances in Cryptology-EUROCRYPT 2014*, pages 423–440. Springer, 2014.
- [DKL09] Yevgeniy Dodis, Yael Tauman Kalai, and Shachar Lovett. On cryptography with auxiliary input. In *Proceedings of the 41st annual ACM Symposium on Theory of Computing*, pages 621–630. ACM, 2009.
- [DORS08] Yevgeniy Dodis, Rafail Ostrovsky, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM Journal on Computing*, 38(1):97–139, 2008.
- [DP08] Stefan Dziembowski and Krzysztof Pietrzak. Leakage-resilient cryptography. In *FOCS*, pages 293–302, 2008.
- [FHN<sup>+</sup>12] Sebastian Faust, Carmit Hazay, Jesper Buus Nielsen, Peter Sebastian Nordholt, and Angela Zottarel. Signature schemes secure against hard-to-invert leakage. In *Advances in Cryptology-ASIACRYPT 2012*, pages 98–115. Springer, 2012.
- [FRR<sup>+</sup>10] Sebastian Faust, Tal Rabin, Leonid Reyzin, Eran Tromer, and Vinod Vaikuntanathan. Protecting circuits from leakage: the computationally-bounded and noisy cases. In *Advances in Cryptology-EUROCRYPT 2010*, pages 135–156. Springer, 2010.
- [Full15] Benjamin Fuller. *Strong Key Derivation from Noisy Sources*. PhD thesis, Boston University, 111 Cummington Ave, Boston, MA 02215, January 2015.
- [GMR88] Shafi Goldwasser, Silvio Micali, and Ronald Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM J. Comput.*, 17(2):281–308, 1988. Preliminary version in FOCS 1984.

- [GST13] Daniel Genkin, Adi Shamir, and Eran Tromer. RSA key extraction via low-bandwidth acoustic cryptanalysis. *IACR Cryptology ePrint Archive*, 2013:857, 2013.
- [GW11] Craig Gentry and Daniel Wichs. Separating succinct non-interactive arguments from all falsifiable assumptions. *STOC. ACM, New York*, pages 99–108, 2011.
- [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.
- [HLR07] Chun-Yuan Hsiao, Chi-Jen Lu, and Leonid Reyzin. Conditional computational entropy, or toward separating pseudoentropy from compressibility. In *EUROCRYPT*, pages 169–186, 2007.
- [ISW03] Yuval Ishai, Amit Sahai, and David Wagner. Private circuits: Securing hardware against probing attacks. In *Advances in Cryptology-CRYPTO 2003*, pages 463–481. Springer, 2003.
- [KJJ99] Paul Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. In *Advances in Cryptology 1999*, pages 388–397. Springer, 1999.
- [Koc96] Paul Kocher. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In *CRYPTO*, pages 104–113, 1996.
- [KPW13] Stephan Krenn, Krzysztof Pietrzak, and Akshay Wadia. A counterexample to the chain rule for conditional hill entropy. In *Theory of Cryptography*, pages 23–39. Springer, 2013.
- [KV09] Jonathan Katz and Vinod Vaikuntanathan. Signature schemes with bounded leakage resilience. In *Advances in Cryptology-ASIACRYPT 2009*, pages 703–720. Springer, 2009.
- [Lam79] Leslie Lamport. Constructing digital signatures from a one-way function. Technical report, Technical Report CSL-98, SRI International, 1979.
- [LMO<sup>+</sup>14] Jake Longo, Daniel Martin, Elisabeth Oswald, Daniel Page, Martijin Stam, and Michael Tunstall. Simulatable leakage: Analysis, pitfalls, and new constructions. In *Advances in Cryptology-ASIACRYPT 2014*, pages 223–242. Springer, 2014.
- [MR04] Silvio Micali and Leonid Reyzin. Physically observable cryptography (extended abstract). In *TCC*, pages 278–296, 2004.
- [NZ93] Noam Nisan and David Zuckerman. Randomness is linear in space. *Journal of Computer and System Sciences*, pages 43–52, 1993.

- [OST06] Dag Arne Osvik, Adi Shamir, and Eran Tromer. Cache attacks and countermeasures: the case of AES. In *Topics in Cryptology–CT-RSA 2006*, pages 1–20. Springer, 2006.
- [Rey11] Leonid Reyzin. Some notions of entropy for cryptography. In *Information Theoretic Security*, pages 138–142. Springer, 2011.
- [Sko14] Maciej Skorski. Modulus computational entropy. In *Information Theoretic Security*, pages 179–199. Springer, 2014.
- [SPY13] François-Xavier Standaert, Olivier Pereira, and Yu Yu. Leakage-resilient symmetric cryptography under empirically verifiable assumptions. In *Advances In Cryptology–CRYPTO 2013*. Springer, 2013.
- [TOS10] Eran Tromer, Dag Arne Osvik, and Adi Shamir. Efficient cache attacks on AES, and countermeasures. *Journal of Cryptology*, 23(1):37–71, 2010.
- [Yao82] Andrew Chi-Chih Yao. Theory and applications of trapdoor functions (extended abstract). In *FOCS*, pages 80–91, 1982.

## A Extending Simulatable Leakage to the Circuit Model

### A.1 Leakage Classes in the Circuit Model

In this section we provide a brief introduction to the circuit leakage model and discuss the applicability of simulatable leakage to the circuit leakage model. The main difference between the circuit and memory leakage model is that the leakage function leaks on a particular implementation of a cryptographic primitive. (In the memory leakage model, leakage is only on the private state.) Most circuit leakage classes assume leakage is “local” to the computation. This makes the leakage class sensitive to the implementation.  $C$  represent a circuit with wires  $C_1, \dots, C_k$  (with the first wires representing the inputs and final wires representing the outputs).

- **Probing** [ISW03]. Let The adversary specifies a subset  $\mathcal{L} \subset C_1, \dots, C_k$  (of bounded size) and sees the values of all wires in  $\mathcal{L}$ .  $\mathcal{L}$  may include some of the secret input, intermediate values, and output values. However, the leakage function is not allowed to compute on parts of the computation simultaneously, the leakage function can only learn the value of individual wires.
- **Computationally Bounded** [FRR<sup>+</sup>10]. Faust et al. [FRR<sup>+</sup>10] hypothesize that leakage can be modeled by low complexity circuits. As an example, they show how to protect circuits against leakage in  $\mathbf{AC}^0$ . They

secret share state, security critically relies on the inability of  $\text{AC}^0$  circuits to compute parity.

- **Noisy** [CJRR99, FRR<sup>+</sup>10]. It is not clear how to precisely determine the computational complexity of a side-channel attack. However, most side-channel attacks are known to contain significant noise. The work of Faust et al. [FRR<sup>+</sup>10] also proposed modeling leakage function as an arbitrary function  $\mathcal{L}$  applied to the secret state with an additive noise term  $\mathcal{N}$ .

Recent work of Duc, Dziembowski, and Faust [DDF14] shows how to simulate a noisy leakage function using a probing leakage function.

## A.2 Adapting Simulatable Leakage to the Circuit Model

We now provide a definition of simulatable leakage that can be used in either the memory or circuit leakage models. As before, we can define both a weak and strong version. We present only a weak version for simplicity.

**Definition A.1.** *Let  $(K, Y)$  be a pair of random variables over  $\chi_1 \times \chi_2$  and  $\mathcal{C}$  be an encoding function such that  $\mathcal{C} : \chi_1 \rightarrow \zeta$ . The randomized map  $\mathcal{L}$  is an  $(\epsilon, s_{sim}, s_{sec}, \mathcal{C})$ -weakly simulatable leakage function if there exists a simulator  $S$  of size at most  $s_{sim}$  such that*

$$\delta^{\mathcal{D}_{s_{sec}}}((Y, \mathcal{L}(\mathcal{C}(K))), (Y, S(Y))) \leq \epsilon.$$

**Notes:** Taking the encoding function to be the identity function yields the memory leakage model. The above definition depends on the encoding function and the leakage class. As an example, for a fixed leakage function, there may be a simulator for only some encoding functions. The work of Ishai, Sahai, and Wagner [ISW03] builds an encoding function where the probing side-channel is simulatable (security of the encoded circuit is shown through simulation).