

Leakage-Resilient Cryptography*

Stefan Dziembowski
University of Rome
La Sapienza

Krzysztof Pietrzak
CWI Amsterdam

Abstract

We construct a stream-cipher SC whose implementation is secure even if a bounded amount of arbitrary (adaptively, adversarially chosen) information about the internal state of SC is leaked during computation of each output block. This captures all possible side-channel attacks on SC where (1) the amount of information leaked in a given period is bounded, but overall can be arbitrary large and (2) “only computation leaks information”.

The construction is based on alternating extraction (used in the intrusion-resilient secret-sharing scheme from FOCS’07). We move this concept to the computational setting by proving a lemma that states that the output of any pseudorandom generator (PRG) has high HILL pseudoentropy (i.e. is indistinguishable from some distribution with high min-entropy) even if arbitrary information about the seed is leaked. The amount of leakage λ that we can tolerate in each step depends on the strength of the underlying PRG, it is at least logarithmic, but can be as large as a constant fraction of the internal state of SC if the PRG is exponentially hard.

1. Introduction

When analyzing the security of a cryptosystem, we can either think of the system as a mathematical object, exactly specifying what kind of access to the functionality a potential adversary has, or try to analyze the security of an actual implementation. Traditionally, cryptographers have mostly considered the former view and analyzed the security of the mathematical object, and it is generally believed that our current knowledge of cryptography suffices to construct schemes that, when modeled in this way, are extremely secure. On a theoretical side, we know how to construct secure primitives under quite weak complexity-theoretic assumptions, for example secret-key encryption can be based

on any one-way function [19]. Also from the practical perspective, the currently used constructions have very strong security properties, e.g. after 30 years of intensive cryptanalytic efforts still the most practical attack on the DES cipher is exhaustive key search.

Side-Channel Attacks. The picture is much more gloomy when the security of *real-life implementations* is considered. The reason is that, when considering the security of an implementation of a cryptosystem, one must take into account the possibility of side-channels, which refers to leakage of any kind of information from the cryptosystem during its execution which cannot be efficiently derived from access to the mathematical object alone. In the last decade many attacks against cryptosystems (still assumed to be sound as mathematical objects) have been found exploiting side-channels like running-time [24], electromagnetic radiation [32, 15], power consumption [25] and many more (see e.g. [33, 30]).¹

A typical countermeasure against this type of attacks is to design hardware that minimizes the leakage of secret data (e.g. by shielding any electromagnetic emissions), or to look for an algorithm-specific solution, for example by masking intermediate variables using randomization (see [30] for a list of relevant papers). The problem with hardware-based solutions is that protection against all possible types of leakage is very hard to achieve [2], if not impossible. On the other hand, most algorithm-specific methods proposed so far are only heuristic and do not offer any formal security proof (we mention some exceptions in Sect.1.1). Moreover, they are ad-hoc in the sense that they protect only against some specific attacks that are known at the moment. This contrasts with the “provable security” approach followed by modern cryptography, where one (1) provides a precise and meaningful model capturing what “secure” means (2) designs a system and (3) proves that this system secure in the general model.

¹This attacks are called *passive* attacks, as all the adversary does is to observe leakage from the computation. In *active* attacks, which are not the subject of this paper, one considers adversaries which intentionally introduce errors in the computation of a cryptodevice [5, 4].

*An extended abstract of this paper appeared at FOCS’08 [13]. This version was last modified February 24, 2010.

Provable Security & Side-Channel Attacks? Clearly, this situation cannot be satisfying from a cryptographic point of view. What are our beautiful provably secure cryptosystems good for, when ultimately their security relies on some ad-hoc countermeasures against side-channel attacks? Despite this, until recently the theory community did not give much attention to this problem. One reason was the perception that side-channels are a practical problem, and theory can only be of limited use to prevent them.

We can roughly distinguish between *continuous* and *bounded* side-channel attacks. A continuous side-channel leaks some information *in each invocation* of the cryptosystem, and thus the amount of leakage can be arbitrary large, in particular, much larger than the internal state of the cryptosystem. Side-channel attacks where the adversary measures some physical leakage like running-time, power-consumption or electromagnetic radiation fall into this class. In contrast, in a *bounded* side-channel attack the amount of leakage the adversary sees is bounded, for example because the adversary can only make a single measurement. An important attacks which falls into this class are cold-boot attacks [18].

There is a substantial body of work (some very recent) giving constructions of cryptosystems which are provably secure against very general *bounded* side-channels. On the other hand, there was almost no progress on provable security against *continuous* leakage predating the conference version of this paper [13]. Notable exceptions are the works on “private-circuits” by Ishai et al. [21] and the and the influential framework of “physically observable cryptography” by Micali and Reyzin [27]. We’ll discuss this and other related work in more detail in Section 1.1. For know, let us just mention that all works on provable security against continuous side-channel attacks either only protect against a very specific side-channel (e.g. [21] only considers “probing attacks”), or, like in [27], require that some functionality can be implemented in a way that already provides security against side-channel attacks.

In this paper we put forward a model of cryptographic computation which gives the adversary the power to learn a bounded amount of arbitrary information about the computation with every invocation. We call cryptographic primitives which are secure in this setting *leakage-resilient*. The motivation for this notion is that it directly implies that any *implementation* of leakage-resilient primitive will be secure in the presence of any side-channel leakage, as long as (1) the amount of information leaked in each invocation is bounded and (2) only parts of the secret state that are actually accessed during an invocation leak in this invocation.

The restriction (1) is inspired by the bounded-storage and bounded-retrieval models and has to best of our knowledge never been used in this context. Restriction (2) has been stated as one of the “axioms” in [27]. Let us stress

that we don’t consider the definition of the model as a contribution, as just mentioned, those ideas have been around for a while. The main contribution is the construction of an actual cryptosystem (a stream-cipher) which is provably secure in this model.

The Model Of Leakage-Resilience. Consider a cryptosystems CS, let state_0 denote its initial state (i.e. the secret key). Side-channel attacks against a cryptosystem CS can be modelled by showing the adversary, besides the regular input/output, the output $f(\text{state}_0)$ of some *leakage function* applied to the secret state.² In the most general setting, which captures all side-channel attacks, we let the adversary choose the leakage function f . Clearly, no keyed cryptosystem can be secure in this setting, as defining $f(\text{state}_0) = \text{state}_0$ the adversary learns the complete secret state, thus the class of leakage-functions must somehow be restricted. Most works on side-channel countermeasures consider leakage-functions capturing very particular side-channels. In contrast, we will only restrict the *amount*, but not the *type* of information that is leaked. Formally, we require that the output range of f is bounded to $\{0, 1\}^\lambda$ where $\lambda \ll |\text{state}_0|$.

To capture continuous attacks, we want to allow the adversary to choose a leakage function many times throughout the lifetime of the device. Technically, this will be done by dividing the execution of the algorithm implementing CS into *rounds*, the adversary can then adaptively choose a leakage function and learn its output when applied to the secret internal state of CS in each of those rounds (let f_j denote the leakage function that she chooses in the j th round, for $j = 1, 2, \dots$). In this paper we will consider stream-ciphers, here a “round” is the computation required to compute some fixed amount of output bits.

Bounded Leakage. Let q be the number of rounds we want our cryptosystem CS to run. At first sight one may think that to hope for any security we would need to assume that $q \cdot \lambda < |\text{state}_0|$, as otherwise the adversary can learn the entire state_0 , by just retrieving in every round λ different bits of it. To avoid this trivial attack one must consider cryptosystems which occasionally update their state. Let state_j denote the state of CS after round j . Without loss of generality we assume that the size of the state stays constant, i.e. $|\text{state}_j| = |\text{state}_{j+1}|$ for any j .

Unfortunately, still no security can be achieved: to see this let $t = \lceil |\text{state}_0|/\lambda \rceil$ and consider $f_j, j \leq t$ where each f_j outputs different λ bits of state_t (note that the function $f_j, j \leq t$ can compute the future state state^t from the current state state^j). After the t th round the adversary has

²Without loss of generality we can assume that the leakage function is applied only to state_0 since all the other internal variables used in computation are deterministic functions of state_0 .

learned the complete state state_t , and no security is possible beyond this point. We call this the *key-precomputation attack*.

Only Computation Leaks Information. Hence, we have to somehow restrict the leakage function if we want security even when the total amount of leaked information is (much) larger than the internal state. The restriction that we will use is that in each round, the leakage function f_j only gets as input the part of the state state_j that is actually accessed in the j th round by CS. We will refer to this restriction as *only computation leaks information* (OCLI for short.)

This term has been coined by Micali and Reyzin [27], it refers to one of their “axioms” that the assume side-channels to satisfy. This axiom is motivated by the fact that basically all side-channels only leak information about the ongoing computation, i.e. the leakage in some round is independent of the memory cells that are not even read in this round.

Every side-channel who satisfies the OCLI axiom (which is a statement about the physical properties of a cryptodevice) is captured by our OCLI restriction (which says something about the modelling of leakage-functions as mathematical objects.) But let us stress that the contrary is not true. In particular, recently Halderman et al. put forward cold-boot attacks [18] in which the adversary can learn a (noisy) version of the *entire* memory even if no computation is going on. This attacks does not adhere the axiom, but (except for some pathological cases) is captured by the model of leakage-resilience. We refer to [?] for a more detailed discussion.

Leakage Resilient Stream-Cipher. The main contribution of this paper is the construction of a stream cipher SC which is provably leakage-resilient. Let us first make precise what leakage-resilience means in the context of stream-ciphers.

Let τ_ℓ denote the data on SC’s memory which is accessed in the ℓ th round, and let K_ℓ denote the output written by SC on its output tape \mathcal{O} in the ℓ th round.

The classical security notion for stream ciphers implies that one cannot distinguish K_ℓ from a random string given $K_1, \dots, K_{\ell-1}$, of course our construction satisfies this notion. But we prove much more, namely that K_ℓ is indistinguishable from random even when not only given $K_0, \dots, K_{\ell-1}$, but additionally $\Lambda_1, \dots, \Lambda_{\ell-1}$ where $\Lambda_j = f_j(\tau_j)$ and each f_j is a function with range $\{0, 1\}^\lambda$ chosen adaptively (as a function of $K_1, \dots, K_{j-1}, \Lambda_1, \dots, \Lambda_{j-1}$) by an adversary. If the adversary also gets Λ_ℓ , we cannot hope that K_ℓ is indistinguishable from random any more, as f_ℓ could for example simply output the λ first bits of K_ℓ . But we can still hope for some security K_ℓ , a natural requirement would be that K_ℓ is hard to predict K_ℓ .

We show something even stronger, namely that it has high HILL-pseudoentropy.

Forward Security. In many settings, it is not enough that K_ℓ is indistinguishable (or unpredictable) given the view of the adversary after round $\ell-1$ as just described, but it should stay indistinguishable even if SC leaks some information *in the future*. In our construction such “forward-security” comes up naturally, as the key K_ℓ is almost independent (in a computational sense) from the state of SC after K_ℓ was output. Precise security definitions are given in Sect. 2.1.

Our Construction. The starting point of our construction is the concept of alternating extraction previously used in the intrusion-resilient secret-sharing scheme from [12]. We move this concept to the computational setting by proving a lemma that states that the output of any PRG has high HILL pseudoentropy (i.e. is indistinguishable from some distribution with high min-entropy) even if a bounded amount of arbitrary information about the seed is leaked. This result (in a more general form) has independently been discovered by [34, 16]. Section 5 of [35] gives an overview over this and related topics, in particular the connection to recent results in number theorem [17].

Our construction can be instantiated with any pseudorandom-generator, and the amount of leakage λ that we can tolerate in each step depends on the strength of the underlying PRG, it is at least logarithmic, but can be as large as a constant fraction of the internal state of SC if the PRG is exponentially secure.

On (Non-)Uniformity. Throughout, we always consider non-uniform adversaries.³ In particular, our stream-cipher is secure against non-uniform adversaries, and we require the PRG used in the construction to be secure against non-uniform adversaries. The only step in the security proof where it matters that we are in a non-uniform setting, is in Section 6, where we use a theorem due to Barak et al. [3] which shows that two notions of pseudoentropy (called HILL and metric-type) are equivalent for circuits. In [3] this equivalence is also proved in a uniform setting, and one could use this to get a stream-cipher secure against uniform adversaries from any PRG secure against uniform adversaries. We will not do so, as for one thing the non-uniform setting is the more interesting one in our context, and moreover the exact security we could get in the uniform setting is

³Recall that a uniform adversary can be modelled as a Turing-machine which as input gets a security parameter, whereas (more powerful) non-uniform adversaries will, for each security parameter, additionally get a different polynomial-length advice string. Equivalently, we can model non-uniform adversaries as a sequence of circuits (indexed by the security parameter).

much worse (due to the security loss in the reduction from [3] in the uniform setting).

1.1. Related work

Alternatives. Let us mention that restricting the only computation leaks information is not the only natural restriction that one could make on the leakage functions to avoid the key-precomputation attack.

One other option might be to allow the state to be refreshed using external randomness. When this randomness is only generated as required.

This option might be difficult to handle for many cryptosystems – including ciphers – for several reasons. For example one must make sure that all legitimate parties get the randomness in each refresh cycle, which means that parties have to be often “online” to keep their key valid, even if they almost never actually use it. Another option is to require that the leakage function is in some very weak complexity class not including the function used for key evolution.

A general theory of side-channel attacks was put forward by Micali and Reyzin [28], who propose a number of “axioms” on which such a theory should be based. In particular they formulate and motivate the assumption that “only computation leaks information”, which we’ll use in this work. As mentioned in the introduction, most published work on securing cryptosystems against side-channel attacks are ad-hoc solutions trying to prevent some particular attack or heuristics coming without security proofs, we mention some notable exceptions below.

Exposure-resilient functions [6, 10, 22] are functions whose output remains secure, even if an adversary can learn the value of some *input* bits, this model has been extensively investigated and very strong results have been obtained.

Ishai et al. [21, 20] consider the more general case of making circuits provably secure [21] and even tamper resistant [20] against adversaries who can read/tamper the value of a bounded number of arbitrary wires in the circuit (and not just the input bits). It is interesting to compare the result from this paper with the approach of Ishai et al. On one hand, their results are generic, in the sense that they provide a method to transform any cryptosystem given as a circuit C into another circuit C_t that is secure against an adversary that can read-off up to t wires, whereas we only construct a particular primitive (a stream-cipher). On the other hand, we prove security against any side-channel attack, whereas Ishai et al. consider the particular case where the adversary can read-off the values of a few individual wires. Moreover Ishai et al. require special gates that can generate random bits, we do not assume any special hardware.

Canetti et al. [7] consider the possibility of secure computation in a setting where perfect deletion of most of the memory is not possible. Although the goal is different,

their model is conceptually very similar to ours: non-perfect deletion of X is modelled by giving an adversary $f(X)$ for a sufficiently compressing function f of its choice. In their setting, the assumption that parts of the state can be perfectly erased is well motivated, unfortunately in our context this would translate to the very unrealistic requirement that some computations can be done perfectly leakage free.

The idea to define the set of leakage functions by restricting the length of function’s output is taken from the bounded-retrieval model [11, 9, 8, 9, 12, 1] which in turn was inspired by the bounded-storage model [26].⁴ Finally let us mention that some constructions of ciphers secure against general leakages were also proposed in the literature, however, their security proofs rely on very strong assumptions like the ideal-cipher model [31], or one-way permutations which do not leak any information at all [28].

1.2. Probability-theoretic preliminaries

We denote with U_n the random variable with distribution uniform over $\{0, 1\}^n$. With $X \sim Y$ we denote that X and Y have the same distribution. Let random variables X_0, X_1 be distributed over some set \mathcal{X} and let Y be a random variable distributed over \mathcal{Y} . Define the *statistical distance between X_0 and X_1* as $\delta(X_0; X_1) = 1/2 \sum_{x \in \mathcal{X}} |P_{X_0}(x) - P_{X_1}(x)|$. Moreover let $\delta(X_0; X_1|Y) := d(X_0, Y; X_1, Y)$ be the *statistical distance between X_0 and X_1 conditioned on Y* . If X is distributed over $\{0, 1\}^n$ then let $d(X) := \delta(X; U_n)$ denote the *statistical distance of X from a uniform distribution (over $\{0, 1\}^n$)*, and let $d(X|Y) := \delta(X; U_n|Y)$ denote the *statistical distance of X from a uniform distribution, given Y* . If $d(X) \leq \epsilon$ then we will say that X is ϵ -close to uniform. We will say that a variable X has *min-entropy k* , denoted $\mathbf{H}_\infty(X) = k$, if $\max_x \Pr[X = x] = 2^{-k}$.

Definition 1 (Extractor) A function $\text{ext} : \{0, 1\}^{k_{\text{ext}}} \times \{0, 1\}^r \rightarrow \{0, 1\}^{m_{\text{ext}}}$ is an $(\epsilon_{\text{ext}}, n_{\text{ext}})$ extractor if for any X with $\mathbf{H}_\infty(X) \geq n_{\text{ext}}$ and $K \sim U_{k_{\text{ext}}}$ we have that $d((\text{ext}(K, X), K) \leq \epsilon_{\text{ext}}$.

⁴The bounded-storage model is limited in its usability by the fact that the secret key must be larger than the memory of a potential adversary, which means in the range of terabytes. In the bounded-retrieval model, the key must only be larger than the amount of data adversary can retrieve without being detected (say, by having a computer-virus send the data from an infected machine), which means in the range of Mega- or Gigabytes. Whereas in our setting the key length depends on the amount of side-channel information that leaks (in one round) from the cryptosystem considered, which (given a reasonable construction) we can assume to be as small as a few (or a few hundred) bits. In particular, unlike the bounded-storage and bounded-retrieval models, our keys need not to be made artificially huge.

2. Security Notions for Stream-Ciphers

A stream cipher is simply a function $SC : \{0, 1\}^s \rightarrow \{0, 1\}^s \times \{0, 1\}^k$. Every initial state $\text{state}_0 \in \{0, 1\}^s$ defined a sequence K_1, K_2, \dots of values as

$$[\text{state}_{i+1}, K_{i+1}] = SC(\text{state}_i) \quad (1)$$

The standard security definition for stream ciphers is to require that the sequence K_1, K_2, \dots is pseudorandom if the initial state state_0 is chosen at random. Alternatively, one can require that for every ℓ' , $K_{\ell'}$ is pseudorandom given $K_1, \dots, K_{\ell'-1}$, which is the notion we define below as it is closer to the leakage-resilience notion we define later. (As shown by Yao [], this doesn't make much of a difference.)

Definition 2 (Indistinguishability without Leakage)

For $\ell \in \mathbb{N}$, $\epsilon = \epsilon(\ell) \in [0, 1]$, $s = s(\ell) \in \mathbb{N}$. SC is a (ℓ, ϵ, s) -secure Stream-cipher if for any distinguisher D of size at most $s - \ell|SC|$ we have

$$|p_0 - p_1| \leq \epsilon$$

where $p_b, b \in \{0, 1\}$ is defined as

$$p_b \stackrel{\text{def}}{=} \Pr_{\text{state}_0} [D(K_1, \dots, K_{\ell-1}, X_b) \rightarrow 1]$$

With $X_0 \stackrel{\text{def}}{=} K_\ell$ and $X_1 = U_k$.

The above definition bounds the ‘‘exact security’’ of SC .

2.1 Leakage-Resilient Stream-Cipher

In the introduction we motivated and informally defined ‘‘leakage-resilient’’ stream-ciphers. In this section we give the formal definition. It will be convenient to define the attacker A as two algorithms (Q, D) . The first Q (for Query) queries SC adaptively choosing leakage functions with every query. The second D (for Distinguish) then tries to distinguish the next output block to be computed by SC from uniformly random.

Recall that state_i is the (secret) internal state of the stream-cipher SC before the $i + 1$ th output block K_{i+1} is computed. We denote with state_i^+ the subset of state_i that is accessed (i.e. read and/or deleted) during the computation of K_{i+1} , with $\text{state}_i^- \stackrel{\text{def}}{=} \text{state}_i \setminus \text{state}_i^+$ we denote the state that is not touched.

Leakage Attacks. The attacker Q can attack SC by learning not only the regular output K_1, K_2, \dots , but with every invocation of SC also λ bits of information $\Lambda_1, \Lambda_2, \dots$ about the state that was accessed. Q must be given as an oracle circuit, where the oracle gates are sequentially ordered. We then consider the following experiment:

1. Set $i := 1$.
2. Q makes a query f_i with describes a circuit with range $\{0, 1\}^\lambda$.
3. Compute $[\text{state}_i, K_i] := SC(\text{state}_{i-1})$.
4. Give K_i and $\Lambda_i := f_i(\text{state}_{i-1}^+)$ to Q .
5. set $i := i + 1$ and go to step 2.

Let $\mathcal{A}_{\lambda, \ell}$ denote adversaries as just described restricted to choose leakage functions with range $\{0, 1\}^\lambda$ and don't stop for at least ℓ rounds.

We denote with $SC \stackrel{\ell}{\rightsquigarrow} Q(\text{state}_0)$ the above experiment, run for ℓ rounds with initial state state_0 . $SC \stackrel{\ell}{\rightsquigarrow} Q$ denotes the same experiment where state_0 is chosen uniformly at random. For any $\ell' \leq \ell$, this experiment defines the random variable

$$\text{view}_{\ell'} \stackrel{\text{def}}{=} [K_1, \dots, K_{\ell'}, \Lambda_1, \dots, \Lambda_{\ell'}]$$

that denotes the view of the adversary of Q after ℓ' rounds in this experiment.

Definition 3 (Indistinguishability with Leakage) For $\ell \in \mathbb{N}$, $\epsilon = \epsilon(\ell) \in [0, 1]$, $s = s(\ell) \in \mathbb{N}$. SC is a (ℓ, ϵ, s) -secure **leakage-resilient** stream-cipher if for any $Q \in \mathcal{A}_{\lambda, \ell-1}$ and $D : \{0, 1\}^{\ell \cdot k + (\ell-1)\lambda} \rightarrow \{0, 1\}$ size

$$|Q| + \ell|SC| + |D| \leq s \quad (2)$$

we have

$$|p_0 - p_1| \leq \epsilon$$

where $p_b, b \in \{0, 1\}$ are defined as

$$p_b \stackrel{\text{def}}{=} \Pr_{\text{state}_0} [D(\text{view}_{\ell-1}, X_b) \rightarrow 1]$$

Where $X_0 \stackrel{\text{def}}{=} K_\ell$ is the next output to be computed by SC and $X_1 \stackrel{\text{def}}{=} U_k$ is uniformly random and $\text{view}_{\ell-1}$ is defined by the random experiment $SC \stackrel{\ell}{\rightsquigarrow} Q$ as described above.

Remark 1 Recall that $|SC|$ denote the size of a circuit computing (one round of) SC . The bound on the size as given by eq.(2) simply means that that size of the entire attack, which means first $SC \stackrel{\ell}{\rightsquigarrow} Q$ and then running D , is bounded by s .

3 The Construction.

We first formally define the construction of our leakage-resilient stream-cipher SC . It uses as building blocks an extractor $\text{ext} : \{0, 1\}^{k_{\text{ext}}} \times \{0, 1\}^r \rightarrow \{0, 1\}^{m_{\text{ext}}}$ and a pseudorandom generator $\text{prg} : \{0, 1\}^{k_{\text{prg}}} \rightarrow \{0, 1\}^r$. In the next section we explain the intuition behind this construction.

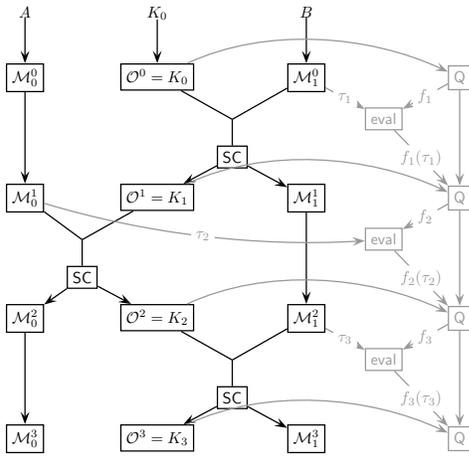


Figure 1. General structure of the random experiment $SC(A, K_0, B) \rightsquigarrow Q$ (the evaluation of SC is black, the attack related part is gray).

state₀: The **initial state** is $state_0 = (A_0, K_0, B_0)$ where $K_0 \in \{0, 1\}^{k_{\text{ext}}}$ and $A_0, B_0 \in \{0, 1\}^r$ are sampled uniformly at random.

state_i: The **state** before the i th output block $K_i \in \{0, 1\}^{k_{\text{ext}}}$ is computed is $state_i = (A_{i-1}, K_{i-1}, B_{i-1})$. (note that the output K_i is kept as part of the internal state.)

state_i⁺: The **active state** $state_i^+$ (i.e. the part of $state_i^+$ that is accessed during the computation of K_{i+1}) is (A_i, K_i) if i is odd and (B_i, K_i) if i is even.

Computation: For odd i , (K_{i+1}, A_{i+1}) is computed from the active state $state_i^+ = (K_i, A_i)$ as

1. $(K_{i+1}, X_{i+1}) := \text{ext}(K_i, A_i)$
2. $A_{i+1} := \text{prg}(X_{i+1})$

The passive state B_i is (by definition) kept unchanged, i.e. $B_{i+1} \stackrel{\text{def}}{=} B_i$. For even i the computation is the same but with the roles of A and B exchanged.

3.1. The Ingredients

The construction is based on “alternating extraction” (AE), which was originally introduced in [12] to construct an intrusion-resilient secret-sharing scheme. Informally, AE is a two-party protocol to compute some value K , which for some $\ell \in \mathbb{N}$, has a very large gap in the communication complexity for ℓ and $\ell - 1$ round protocols.

Alternating Extraction. Let $\text{ext} : \{0, 1\}^{k_{\text{ext}}} \times \{0, 1\}^r \rightarrow \{0, 1\}^k$ be an $(\epsilon_{\text{ext}}, n_{\text{ext}})$ -extractor (cf. Def. 1). Consider some uniformly random $A, B \in \{0, 1\}^r$ and some random $K_0 \in \{0, 1\}^k$. As illustrated in Fig. 3 in Sect. 5, let K_1, K_2, \dots be computed as $K_i = \text{ext}(K_{i-1}^{\text{next}}, C_i)$ (where K^{next} denotes the k_{ext} first bits of K and $C_i = B$ if i is odd and $C_i = A$ otherwise). So the K_i ’s are computed by alternately extracting from A and B . It is not hard to show that $K_i = \text{ext}(K_{i-1}^{\text{next}}, C_i)$ is $i\epsilon_{\text{ext}}$ close to uniformly random given K_0, \dots, K_{i-1} while C_i has still enough min-entropy for our extractor (i.e. $\mathbf{H}_\infty(C_i | K_1, \dots, K_{i-1}) \geq n_{\text{ext}}$).

As shown in [12], the key K_i is even close to uniformly random when not only given K_1, \dots, K_{i-1} but also some values $f_1(C_1), \dots, f_{i-1}(C_{i-1})$ for arbitrary functions f_i as long as C_i has min-entropy at least n_{ext} (conditioned on K_0, \dots, K_{i-1} , and $f_1(C_1), \dots, f_{i-1}(C_{i-1})$).

Consider a “stream cipher” $SC^*(A, B, K_0)$ which outputs K_1, K_2, \dots computed as described above, and an adversary Q which, before K_i is computed, can adaptively choose a function f_i and then gets $K_i, f_i(C_i)$.⁵ As explained in the previous paragraph, we can give the following security guarantee for SC^* : as long as the min-entropy of C_i is at least n_{ext} (given the adversary’s view), the next output K_i is close to uniformly random (given the view of the adversary so far).

Pseudoentropy. The stream cipher SC^* just described is not very useful, as it only provides security (in the sense of Definition 3) as long as the output (i.e. the K_i ’s plus the leaked information) is shorter (by at least n_{ext} bits) than the initial key.

To get security beyond that bound, we “refresh” the values A and B after extracting from them (we denote the values after the i th round with A_i, B_i .) In round i (we assume i is odd, otherwise replace the role of A and B) we extract $(K, X_i) = \text{ext}(K_{i-1}^{\text{next}}, B_{i-1})$, and use X_i to compute the fresh $B_i := \text{prg}(X_i)$ using a pseudorandom generator prg as illustrated in Fig. 2. If at the beginning of the i th round B_{i-1} has min-entropy at least n_{ext} (given the adversary’s view), K_{i-1}^{next} is pseudorandom (given B_i) and we assume that during this i th round no information is leaked, then X_i , and thus also $B_i = \text{prg}(X_i)$ is pseudorandom given the view of the adversary.

Of course assuming that the refreshing phase does not leak any information is completely unjustified, and we do not want to make such an assumption. As we give $\Lambda_i = f_i(B_i)$ to the adversary, we cannot hope for B_i to be pseudorandom (just consider the case where $f_i(B_i)$ are the λ first bits of B_i). Fortunately, B_i needs not to be (pseudo)random to apply alternating extraction, all we need is that B_i has high min-entropy. Of course $B_i = \text{prg}(X_i)$

⁵As K_{i-1} can be hard-coded into f_i , this function has access to all the data accessed during the computation of $K_i = \text{ext}(K_{i-1}^{\text{next}}, C_i)$

cannot have more min-entropy than X_i , but as we consider computationally bounded adversaries, it is enough if B_i is indistinguishable from some distribution with high min-entropy. A random variable which is computationally indistinguishable from some variable with min-entropy k is said to have HILL-pseudoentropy k . It is not hard to see that a pseudorandom value B_i has high HILL-pseudoentropy when given $f_i(B_i)$ for some efficient function f_i , but this is not enough for our application, as the leakage function f_i is given access to B_{i-1} (and not just B_i), from which it can compute the seed X_i used to compute $B_i = \text{prg}(X_i)$. We will prove (Lemma 3) that for any pseudorandom generator prg , the output of $\text{prg}(X)$ on a random seed X has high HILL-pseudoentropy even if some function (with sufficiently short output) of X (and not only $\text{prg}(X)$) is leaked.

Using this lemma, we can prove that refreshing using a PRG as just described actually works, and will result in a “fresh” value B_i (or A_i for even i) having high HILL-pseudoentropy.

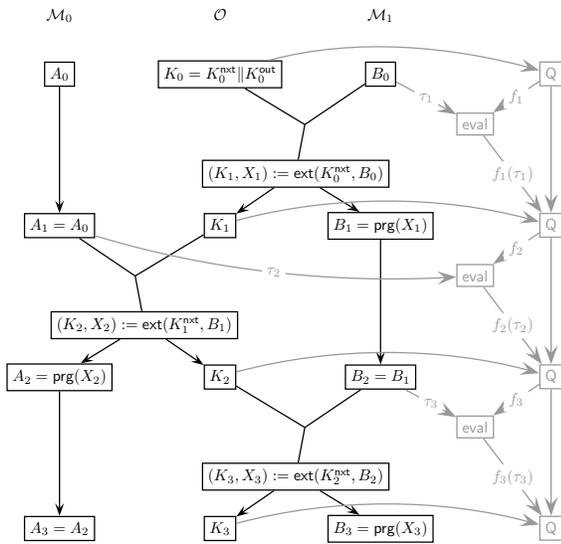


Figure 2. Illustration of the random experiment $\text{SC}(A_0, B_0, K_0) \xrightarrow{3} \text{Q}$ for the stream cipher SC.

4. Security of SC

Theorem 1 (Security of SC) Consider the construction $\text{SC} \stackrel{\text{def}}{=} \text{SC}[\text{ext}, \text{prg}]$ based on

- An $(\epsilon_{\text{ext}}, m_{\text{ext}})$ -extractor $\text{ext} : \{0, 1\}^{k_{\text{ext}}} \times \{0, 1\}^r \rightarrow \{0, 1\}^{m_{\text{ext}}}$.
- An $(\epsilon_{\text{prg}}, s_{\text{prg}})$ -secure pseudorandom generator $\text{prg} : \{0, 1\}^{k_{\text{prg}}} \rightarrow \{0, 1\}^r$.

For any $\ell \in \mathbb{N}$ and consider any $\epsilon_{\text{HILL}} > 0$ and let $\hat{s} \approx \epsilon_{\text{HILL}}^2 s_{\text{prg}} / 8r$.⁶ Consider any $\epsilon_{\text{gap}} > 0, \Delta > 0$ where

$$\epsilon_{\text{prg}} \leq \frac{\epsilon_{\text{gap}}^2}{2^\lambda} - 2^{-\Delta}, \quad n_{\text{ext}} \leq r - \Delta - (\lambda + m_{\text{ext}}) - 2 \log(1/\epsilon_{\text{gap}}) \quad (3)$$

Define $\delta_\ell \stackrel{\text{def}}{=} \ell^2 \cdot \epsilon_{\text{ext}} + 2\ell \cdot \epsilon_{\text{HILL}} + 4\ell \cdot \epsilon_{\text{gap}}$. Then SC is an $(\ell, \delta_\ell, \hat{s})$ -secure leakage-resilient stream-cipher.

The proof of Theorem 1 is split in three parts. The first part in Section 5 on alternating extraction is information theoretic and uses ideas from the intrusion-resilient secret-sharing scheme from [12]. In the second part (Section 7) we revisit some notions and results on computational pseudoentropy. We then prove that the output of any pseudorandom generator has high HILL pseudoentropy even if information about the seed is leaked. In Section 7 we prove Theorem 1 by using the result from Section 6 to get a computational version of alternating extraction from Section 5.

How Much Leakage can we Tolerate? The amount of leakage λ we can tolerate is bounded by (3) as $\epsilon_{\text{prg}} \leq \epsilon_{\text{gap}}^2 / 2^\lambda - 2^{-\Delta}$. For concreteness, assume we set Δ such that $2^{-\Delta} \leq \epsilon_{\text{prg}} / 2$ and $\epsilon_{\text{gap}} \geq \sqrt[4]{\epsilon_{\text{prg}} / 4}$, then we can set

$$\lambda = \left\lfloor \frac{\log \epsilon_{\text{prg}}^{-1}}{2} \right\rfloor$$

To see what this means it is convenient to take an asymptotic viewpoint and think of SC as a family of stream ciphers indexed by a security parameter which we identify with k_{prg} , i.e. the input length to prg . If prg is secure against polynomial-size circuits, then $\epsilon_{\text{prg}} = 2^{-\omega(\log k_{\text{prg}})}$ (and thus $\lambda \in \omega(\log k_{\text{prg}})$), and if prg is secure against exponential size circuits, then $\epsilon_{\text{prg}} = 2^{-\Theta(k_{\text{prg}})}$ (and $\lambda \in \Theta(k_{\text{prg}})$).

Already the $\lambda \in \omega(\log k_{\text{prg}})$ case covers quite a large class of real-life attacks. In particular many attacks based on measuring the power consumption result in logarithmic-size leakages, e.g. in a so-called *Hamming weight attack* (see e.g. [23]) the adversary just learns the number of wires carrying the bit 1. Of course this value is of logarithmic length in the size of the circuit, and hence also in k_{prg} .

In the case $\lambda \in \Theta(k_{\text{prg}})$ (i.e. if prg is exponentially hard) one can leak even a constant fraction of the entire state of SC.

5. Random Keys by Alternating Extraction

In this section we state an information theoretic result which is very similar to the main technical lemma used in the security proof of the intrusion-resilient secret-sharing scheme from [12], a proof appears in [14].

⁶See Lemma 2 as to what \hat{s} exactly is.

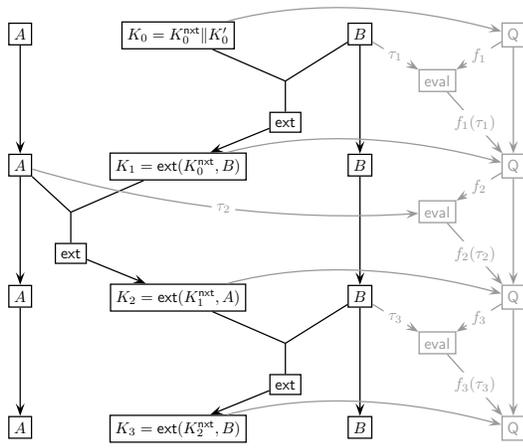


Figure 3. The “alternating extraction” random experiment $SC^*(A, B, K_0) \xrightarrow{3} Q$ as considered in Lemma 1.

Basically, we consider the random experiment $SC \xrightarrow{\ell} Q$ but without the refreshing. For this let SC^* denoted the construction SC but where A and B are never replaced: thus in the random experiment $SC^*(A, B, K_0) \xrightarrow{\ell} Q$ where $Q \in \mathcal{A}_\lambda$, in the j th round Q chooses a function $f_j : \{0, 1\}^r \rightarrow \{0, 1\}^\lambda$ and as output gets $K_j = \text{ext}(K_{j-1}^{\text{nxt}}, \tau_j)$ and $\Lambda_j = f_j(\tau_j)$ where $\tau_j = B$ if j is odd and $\tau_j = A$ otherwise.

As Q attacks SC^* , she learns information on A and B , and thus the min-entropy of A and B degrades. We show that as long as the min-entropy of A and B is high enough (which means more than n_{ext} as required by the extractor ext), the next key K_j to be output is close to uniformly random when given the view after K_{j-1} has been computed.

Lemma 1 belows similar to Lemma 8 from [12] (for the special case of two players).

Lemma 1 (Alternating Extraction) *Let $\text{ext} : \{0, 1\}^{k_{\text{ext}}} \times \{0, 1\}^r \rightarrow \{0, 1\}^{n_{\text{ext}}}$ be an $(\epsilon_{\text{ext}}, n_{\text{ext}})$ -extractor. Let $A, B \in \{0, 1\}^r$ and $K_0 \in \{0, 1\}^k$ be random variables where A and B are independent and*

$$d(K_0|B) \leq \epsilon_0 \quad \mathbf{H}_\infty(A) \geq r - \Delta \quad \mathbf{H}_\infty(B) \geq r - \Delta,$$

Consider any $\lambda, \Delta, r \geq 0$ and $1 \geq \epsilon_{\text{gap}} > 0$ which satisfy

$$n_{\text{ext}} \leq r - \Delta - \lceil \ell/2 \rceil (\lambda + m_{\text{ext}}) - \log(1/\epsilon_{\text{gap}}).$$

Consider any adversary $Q \in \mathcal{A}_\lambda$ and the random experiment $SC^*(A, B, K_0) \xrightarrow{\ell} Q$. Recall that $\text{view}_\ell = [K_0, \dots, K_\ell, \Lambda_1, \dots, \Lambda_\ell]$ and $\tau_\ell = B$ if ℓ is odd and $\tau_\ell = A$ otherwise. We have

$$d(K_{\ell+1}|\text{view}_\ell, \tau_\ell) \leq (\ell + 1)\epsilon_{\text{ext}} + 2\epsilon_{\text{gap}} + \epsilon_0,$$

i.e. given τ_ℓ and the view of Q after the computation of K_ℓ , the next key $K_{\ell+1} = \text{ext}(K_\ell, \tau_\ell)$ to be output by SC is $(\ell\epsilon_{\text{ext}} + 2\epsilon_{\text{gap}} + \epsilon_0)$ -close to uniformly random.

6. Pseudoentropy

In this section we will prove that the output of a PRG has high HILL-pseudoentropy even if some function of the seed is leaked. We first prove this result for a weaker notion of pseudoentropy called “metric-type”, and then use the equivalence of metric-type and HILL-pseudoentropy (Lemma 2) to get our lower bound for HILL-pseudoentropy.

Basic Definitions We denote with $\delta^D(X; Y)$ the advantage of a circuit D in distinguishing the random variables X, Y , i.e.: $\delta^D(X; Y) \stackrel{\text{def}}{=} |E[D(X)] - E[D(Y)]|$. Let \mathcal{D}_s denote the class of all probabilistic circuits of size s with binary output $\{0, 1\}$ and \mathcal{D}_s^* denote the class of deterministic circuits with output range $[0, 1]$.

With $\delta_s(X; Y)$ ($\delta_s^*(X; Y)$) we denote $\max_{D \in \mathcal{D}_s} \delta^D(X; Y)$ where the maximum is over $D \in \mathcal{D}_s$ ($D \in \mathcal{D}_s^*$). For a random variable X over $\{0, 1\}^z$, $d_s(X) \stackrel{\text{def}}{=} \delta_s(X; U_z)$ and $d_s^*(X) \stackrel{\text{def}}{=} \delta_s^*(X; U_z)$.

Definition 4 (Pseudorandom Generator) *A function $\text{prg} : \{0, 1\}^n \rightarrow \{0, 1\}^m$ is a (δ, s) -secure pseudorandom generator (PRG) if $d_s(\text{prg}(U_n)) \leq \delta$.*

Definition 5 (HILL pseudoentropy [19, 3]) *We say X has HILL pseudoentropy k , denoted by $\mathbf{H}_{\epsilon, s}^{\text{HILL}}(X) \geq k$, if there exists a distribution Y where $\mathbf{H}_\infty(Y) \geq k$ and $\delta_s(X, Y) \leq \epsilon$. $\mathbf{H}_{\epsilon, s}^{\text{HILL}}$ is defined analogously but using δ_s^* instead δ_s .*

The above definition requires that there exists a distribution Y with high min-entropy that is indistinguishable from X by all distinguishers. One can also consider a notion where the quantifiers are exchanged, i.e. to allow the distribution to depend on the distinguisher.

Definition 6 (Metric-type pseudoentropy [3]) *We say X has metric-type pseudoentropy k , denoted $\mathbf{H}_{\epsilon, s}^{\text{Metric}}(X) \geq k$, if for every circuit $D \in \mathcal{D}_s$ there exists a distribution Y with $\mathbf{H}_\infty(Y) \geq k$ and $\delta^D(X, Y) \leq \epsilon$. $\mathbf{H}_{\epsilon, s}^{\text{Metric}}$ is defined analogously using \mathcal{D}_s^* instead \mathcal{D}_s .*

Barak et al. [3] use the von Neumann’s min-max theorem [29] to prove the equivalence of \mathbf{H}^{HILL} and $\mathbf{H}^{\text{Metric}}$.

Lemma 2 [Thm.5.2 from [3]] *Let X be a distribution over $\{0, 1\}^n$. For every $\epsilon, \epsilon_{\text{HILL}} > 0$ and k , if $\mathbf{H}_{\epsilon, s}^{\text{Metric}}(X) \geq k$ then $\mathbf{H}_{\epsilon + \epsilon_{\text{HILL}}, \hat{s}}^{\text{HILL}}(X) \geq k$ where $s \in O(n\hat{s}/\epsilon_{\text{HILL}}^2)$ or equivalently $\hat{s} \in \Omega(\epsilon_{\text{HILL}}^2 s/n)$. More precisely (by inspection of the proof of Thm.5.2 in [3]) $s \leq 8n\hat{s}/\epsilon_{\text{HILL}}^2 - \zeta$ where ζ is the size of a circuit needed to compute the majority of $8n/\epsilon_{\text{HILL}}^2$ bits.*

6.1. Pseudoentropy of a PRG

By the following lemma, the output of a PRG has high metric-type pseudoentropy (and thus by Lemma 2 also high HILL-pseudoentropy) even if some function of its input is leaked.

Lemma 3 (Metric/HILL Pseudoentropy of a PRG) *Let $\text{prg} : \{0, 1\}^n \rightarrow \{0, 1\}^m$ and $f : \{0, 1\}^n \rightarrow \{0, 1\}^\lambda$ (where $1 \leq \lambda < n < m$) be any functions. If prg is a $(\epsilon_{\text{prg}}, s_{\text{prg}})$ -secure pseudorandom-generator, then for any $\epsilon, \Delta > 0$ satisfying $\epsilon_{\text{prg}} \leq \frac{\epsilon^2}{2^\lambda} - 2^{-\Delta}$, we have with $X \sim U_n$*

$$\Pr_{y:=f(X)} [\mathbf{H}_{\epsilon, s_{\text{prg}}}^{\text{Metric}}(\text{prg}(X)|f(X) = y) \geq m - \Delta] \geq 1 - \epsilon \quad (4)$$

and for any $\epsilon_{\text{HILL}} > 0$

$$\Pr_{y:=f(X)} [\mathbf{H}_{\epsilon + \epsilon_{\text{HILL}}, \hat{s}}^{\text{HILL}}(\text{prg}(X)|f(X) = y) \geq m - \Delta] \geq 1 - \epsilon \quad (5)$$

where $\hat{s} \approx \epsilon_{\text{HILL}}^2 s_{\text{prg}} / 8m$.

Proof: Eq. (5) follows from (4) by Lemma 2. To prove (4) assume for contradiction that it does not hold. Hence, by Def. 6, there exists a subset

$$\mathcal{S} \subseteq \{0, 1\}^\lambda \quad \text{where} \quad \Pr[f(U_n) \in \mathcal{S}] > \epsilon \quad (6)$$

such that for each $a \in \mathcal{S}$ there exists a distinguisher $D_a \in \mathcal{D}_{s_{\text{prg}}}^*$ such that for every random variable Z with $\mathbf{H}_\infty(Z) \geq m - \Delta$ we have (again $X \sim U_n$)

$$|E[D_a(Z)] - E[D_a(\text{prg}(X))|f(X) = a]| \geq \epsilon \quad (7)$$

Consider some $a \in \mathcal{S}$ for which

$$\Pr[f(U_n) = a] > 2^{-\lambda} \cdot \epsilon \quad (8)$$

Such an a exists by (6) and as $|\mathcal{S}| = 2^\lambda$. Let Z^- and Z^+ distributions of min-entropy $m - \Delta$ minimizing $E[D_a(Z^-)]$ and maximizing $E[D_a(Z^+)]$ respectively. Let $\beta^- \stackrel{\text{def}}{=} E[D_a(Z^-)]$, $\beta^+ \stackrel{\text{def}}{=} E[D_a(Z^+)]$ and $\beta \stackrel{\text{def}}{=} E[D_a(\text{prg}(X))|f(X) = a]$.

Claim 1 *Either $\beta^+ - \epsilon \leq \beta$ or $\beta^- + \epsilon \geq \beta$.*

Proof of Claim: Assume the statement of the Claim does not hold, then we can define a distribution Z with $\mathbf{H}_\infty(Z) \geq m - \Delta$ as a convex combination of Z^- and Z^+ where $|E[D_a(Z)] - \beta| < \epsilon$ contradicting (7). \triangle

For the rest of the proof we will assume that the first case $\beta^+ - \epsilon \leq \beta$ guaranteed to hold by the above claim is true (the proof for the second case is symmetric).

Claim 2 $\Pr[D_a(U_m) > \beta^+] < 2^{-\Delta}$

Proof of Claim: Assume for contradiction $\Pr[D_a(U_m) > \beta^+] \geq 2^{-\Delta}$ and let Z be uniform over $S := \{s; D_a(s) > \beta^+\}$. As $|S| \geq 2^{m-\Delta}$ we have $\mathbf{H}_\infty(Z) > m - \Delta$. Moreover by definition of S we have $E[D_a(Z)] > \beta^+$ contradicting the definition of Z^+ . \triangle

Claim 3 $\Pr[D_a(\text{prg}(X))|f(X) = a] > \beta^+] \geq \epsilon$

Proof of Claim: The above follows by Markov using $E[D_a(\text{prg}(X))|f(X) = a] = \beta \geq \beta^+ - \epsilon$. \triangle

Claim 4 $\Pr[D_a(\text{prg}(X)) > \beta^+] \geq \epsilon^2 / 2^\lambda$

Proof of Claim: This follows from Claim 3 and (8). \triangle Let $D'_a(\cdot)$ be a distinguisher where $D'_a(x)$ outputs 1 if $D_a(x) > \beta^+$ and 0 otherwise. By Claims 2 and 4 the advantage of D'_a for U_m and $\text{prg}(U_n)$ is at least (in the last step we use our assumption on ϵ_{prg} from the statement of the Lemma)

$$\begin{aligned} E[D'_a(\text{prg}(U_n))] - E[D'_a(U_m)] &= \\ \Pr[D'_a(\text{prg}(U_n)) = 1] - \Pr[D'_a(U_m) = 1] &> \\ \frac{\epsilon^2}{2^\lambda} - 2^{-\Delta} &\geq \epsilon_{\text{prg}} \end{aligned}$$

which contradicts the $(\epsilon_{\text{prg}}, s_{\text{prg}})$ -security of prg . \square

7 Putting Things Together

In this section we prove Theorem 1 which states that SC is a leakage-resilient stream-ciphec (cf. Definition 3).

Consider Q, D as in the statement of the Theorem. As defined in Section 2.1, let

$$\text{view}_{\ell-1} = [K_0, \dots, K_{\ell-1}, \Lambda_1, \dots, \Lambda_{\ell-1}]$$

denote the view of Q in the random experiment $\text{SC}(A_0, B_0, K_0) \stackrel{\ell-1}{\rightsquigarrow} Q$, where A_0, B_0, K_0 are uniformly random. We must prove that K_ℓ is pseudorandom given $\text{view}_{\ell-1}$, more precisely, we must bound $|p_{\text{real}} - p_{\text{rand}}|$ where

$$p_{\text{real}} = \Pr_{\text{state}_0} [D(\text{view}_{\ell-1}, K_\ell) \rightarrow 1]$$

and

$$p_{\text{rand}} = \Pr_{\text{state}_0} [D(\text{view}_{\ell-1}, U_k) \rightarrow 1]$$

The Hybrid Experiments. It will be convenient to define

$$C_i \stackrel{\text{def}}{=} A_i \text{ if } i \text{ is odd, and } C_i \stackrel{\text{def}}{=} B_i \text{ if } i \text{ is even.}$$

We consider a sequence $G_0, \dots, G_{\ell-1}$ of hybrid experiments, where G_0 is the original experiment $(A_0, B_0, K_0) \stackrel{\ell-1}{\rightsquigarrow} Q$. The game G_i is defined as G_{i-1} except that that we replace C_i with some random variable

that has high min-entropy. This replacement is done right after C_i has been computed. As C_i has (almost certainly) high HILL-entropy, we can replace it so that the two experiments G_i and G_{i-1} are indistinguishable. This then implies that also G_0 (the original experiment) and $G_{\ell-1}$ are indistinguishable. In $G_{\ell-1}$ the variable K_ℓ is statistically close to uniformly random, thus here no D can distinguish it for information theoretic reasons. As G_0 is indistinguishable from $G_{\ell-1}$, and in $G_{\ell-1}$ the last output K_ℓ is close to uniform, it follows that K_ℓ is also pseudorandom in the original experiment G_0 .

We will use an additional superscript i to denote that a variable in the experiment G_i is considered, in particular

$$\text{view}_j^i = [K_0^i, \dots, K_j^i, \Lambda_1^i, \dots, \Lambda_j^i]$$

is the view of Q in the experiment G_i after j rounds. For $i = 1, \dots, \ell$ we define

$$p_{\text{real}}^i = \Pr_{\text{state}_0} [\text{D}(\text{view}_{\ell-1}^i, K_\ell^i) \rightarrow 1]$$

and

$$p_{\text{rand}}^i = \Pr_{\text{state}_0} [\text{D}(\text{view}_{\ell-1}^i, U_k) \rightarrow 1]$$

Using the triangle inequality (i.e. $|a - c| \leq |a - b| + |b - c|$ for any $a, b, c \in \mathbb{R}$) we upper bound $|p_{\text{real}} - p_{\text{rand}}|$ as

$$|p_{\text{real}}^0 - p_{\text{rand}}^0| \leq (9)$$

$$\sum_{i=1}^{\ell-1} |p_{\text{real}}^i - p_{\text{real}}^{i-1}| + \sum_{i=1}^{\ell-1} |p_{\text{rand}}^i - p_{\text{rand}}^{i-1}| + |p_{\text{real}}^{\ell-1} - p_{\text{rand}}^{\ell-1}| \quad (10)$$

Below we formally define the games G_i . By induction on i , we will show that the following invariants hold in the game G_i :

1. A_j^i and B_j^i are independent given view_j^i for any $0 \leq j \leq \ell - 1$.
2. $\Pr[\mathbf{H}_\infty(C_{i-1}^i | \text{view}_{i-1}^i) \geq n - \Delta] = 1$.
3. K_{i-1}^i is $(i - 1) \cdot \epsilon_{\text{ext}}$ -close to uniform given view_{i-2}^i and C_{i-2} .

The Games. G_0 is the original experiment $(A_0, B_0, K_0) \stackrel{\ell-1}{\rightsquigarrow} \text{Q}$. For concreteness, we now explain how G_i is derived from G_{i-1} for even i (for odd i , one just must exchange the A 's with B 's everywhere). Up to round $i - 1$ the games G_{i-1} and G_i are defined exactly the same.

In round i of game G_{i-1} one computes $(K_{i-1}^{i-1}, X_i^{i-1}) \leftarrow \text{ext}(K_{i-1}^{i-1}, B_{i-1}^{i-1})$ and then $A_i^{i-1} = \text{prg}(X_i^{i-1})$ (cf. Figure 2). By the following claim, this A_i^{i-1} has high pseudoentropy.

Claim 5

$$\begin{aligned} & \Pr_{\text{view}_{i-1}^{i-1}} [\mathbf{H}_{\epsilon_{\text{gap}} + \epsilon_{\text{HILL}}, \hat{s}}^{\text{HILL}}(A_i^{i-1} | \text{view}_{i-1}^{i-1}, B_{i-1}^{i-1}) \geq r - \Delta] \\ & \geq 1 - \epsilon_{\text{gap}} - i \cdot \epsilon_{\text{ext}} \end{aligned}$$

Proof of Claim: By invariant 2. we have

$$\mathbf{H}_\infty(B_{i-1}^{i-1} | \text{view}_{i-1}^{i-1}) \geq n - \Delta \quad (11)$$

and by invariant 3.

$$d(K_{i-1}^{i-1} | \text{view}_{i-1}^{i-1}, B_{i-1}^{i-1}) \leq (i - 1) \epsilon_{\text{ext}} \quad (12)$$

As $(K_i^{i-1}, X_i^{i-1}) \leftarrow \text{ext}(K_{i-1}^{i-1}, B_{i-1}^{i-1})$, it follows from the security of our extractor and eq.(11) and (12) that

$$d([K_i^{i-1}, X_i^{i-1}] | \text{view}_{i-1}^{i-1}) \leq i \cdot \epsilon_{\text{ext}} \quad (13)$$

This directly implies that $A_i^{i-1} = \text{prg}(X_i^{i-1})$ is pseudorandom given view_{i-1}^{i-1} and K_i^{i-1} . A_i^{i-1} is even pseudorandom when additionally conditioned on $B_{i-1}^{i-1} = B_i^{i-1}$, as by the first invariant this variable is independent of A_i^{i-1} . Further, by Lemma 3, A_i^{i-1} has high HILL-pseudoentropy given the leakage of the i 'th round, i.e.

$$\begin{aligned} & \Pr_{\text{view}_i^{i-1}} [\mathbf{H}_{\epsilon_{\text{gap}} + \epsilon_{\text{HILL}}, \hat{s}}^{\text{HILL}}(A_i^{i-1} | \text{view}_i^{i-1}, B_{i-1}^{i-1}) \geq r - \Delta] \\ & \geq 1 - \epsilon_{\text{gap}} - i \cdot \epsilon_{\text{ext}} \end{aligned}$$

As stated in the claim. \triangle

Game G_i is run exactly as game G_{i-1} up to the point where K_i^i is computed. Thus, by the above claim, we have

$$\begin{aligned} & \Pr_{\text{view}_i^i} [\mathbf{H}_{\epsilon_{\text{gap}} + \epsilon_{\text{HILL}}, \hat{s}}^{\text{HILL}}(A_i^i | \text{view}_i^i, B_{i-1}^i) \geq r - \Delta] \\ & \geq 1 - \epsilon_{\text{gap}} - i \cdot \epsilon_{\text{ext}} \end{aligned}$$

In game G_i , we then replace A_i^i with a random variable \tilde{A}_i^i with min-entropy $n - \Delta$. By the above equation, we can choose this \tilde{A}_i^i such that it is indistinguishable, more precisely

$$\Pr_{\text{view}_i^i} [\delta_{\hat{s}}(A_i^i, \tilde{A}_i^i | \text{view}_i^i, B_{i-1}^i) \leq \epsilon_{\text{gap}} + \epsilon_{\text{HILL}}] \geq 1 - \epsilon_{\text{gap}} - i \cdot \epsilon_{\text{ext}} \quad (14)$$

As G_{i-1} and G_i differ only by the fact that in G_i we replace A_i^i with a random variable that is computationally indistinguishable, it follows that also $\text{D}(\text{view}_{\ell-1}^{i-1})$ and $\text{D}(\text{view}_{\ell-1}^i)$ cannot be distinguished, which, as we'll explain in more detail below, implies

$$|p_{\text{real}}^{i-1} - p_{\text{real}}^i| \leq 2 \cdot \epsilon_{\text{gap}} + \epsilon_{\text{HILL}} + i \cdot \epsilon_{\text{ext}} \quad (15)$$

$$|p_{\text{rand}}^{i-1} - p_{\text{rand}}^i| \leq 2 \cdot \epsilon_{\text{gap}} + \epsilon_{\text{HILL}} + i \cdot \epsilon_{\text{ext}} \quad (16)$$

To prove that (14) implies (15) and (16) we exploits two things. First, that the size of the entire random experiment considered (i.e. first having Q querying the cipher and

then running D on the view) is less than the size \hat{s} against which A_i^i and \tilde{A}_i^i are proven secure (cf. eq.(14)), and second, that A_i^i and \tilde{A}_i^i are indistinguishable given all other random variables which exist in the experiment at the time-point where the replacement happens (i.e. we conditioned on $[\text{view}_i^i, B_{i-1}^i]$.) Technically, (15) and (16) follow from (14) by applying the “replacement lemma” Lemma 4 which we state and prove below.

We can bound the last term of eq.(9) using eq.(12) which states that K_ℓ^ℓ is $\ell \cdot \epsilon_{\text{ext}}$ close (statistically) to uniform (given $\text{view}_{\ell-1}^\ell$) which implies

$$|p_{\text{real}}^{\ell-1} - p_{\text{rand}}^{\ell-1}| \leq \ell \cdot \epsilon_{\text{ext}} \quad (17)$$

Using the bound from (15) and (17) in eq.(9) we get

$$|p_{\text{real}}^0 - p_{\text{rand}}^0| \leq \ell^2 \cdot \epsilon_{\text{ext}} + 2\ell \cdot \epsilon_{\text{HILL}} + 4\ell \cdot \epsilon_{\text{gap}}$$

The Replacement Lemma. Let A and V be random variables where A has k bits pseudoentropy conditioned on any v in the support of V

$$\mathbf{H}_{\epsilon, s}^{\text{HILL}}(A|V = v) = k. \quad (18)$$

Recall that this means that there exists a random variable \tilde{A} with

$$\mathbf{H}_\infty(\tilde{A}|V = v) = k \quad (19)$$

that is indistinguishable from A , i.e.

$$\delta_s(A, \tilde{A}|V = v) \leq \epsilon. \quad (20)$$

Lemma 4 For any function ψ and $A, \tilde{A}, V = v$ satisfying (18)-(20)

$$\delta_{s-|\psi|}(\psi(A, v), \psi(\tilde{A}, v)) \leq \epsilon$$

Proof: For contradiction assume there exists a distinguisher D of size at most $s - |\psi|$ where

$$\delta^D(\psi(A, v), \psi(\tilde{A}, v)) > \epsilon \quad (21)$$

Now let D' be a distinguisher of size $s - |\psi| + |\psi| = s$ where $D'(x)$ computes and outputs $D(\phi(x, b, v))$. We get with eq.(21) in the last step

$$\delta^{D'}(A, \tilde{A}) = \delta^D(\psi(A, v), \psi(\tilde{A}, v)) > \epsilon$$

which contradicts (20). \square

Acknowledgements

We're grateful to Salil Vadhan for pointing out that Theorem 5.2 from [3] (Lemma 2 in this paper) does not hold for deterministic circuits with binary output (as we wrongfully assumed in the FOCS'08 proceedings version of this paper), but for circuits with range $[0, 1]$. The proof of Lemma 3 in this version is adapted to take this into account.

References

- [1] Joel Alwen, Yevgeniy Dodis, and Daniel Wichs. Leakage resilient public-key cryptography in the bounded retrieval model. In *CRYPTO*, 2009.
- [2] Ross Anderson and Markus Kuhn. Tamper resistance: a cautionary note. In *WOEC'96: Proceedings of the 2nd conference on Proceedings of the Second USENIX Workshop on Electronic Commerce*, pages 1–11, Berkeley, CA, USA, 1996. USENIX Association.
- [3] Boaz Barak, Ronen Shaltiel, and Avi Wigderson. Computational analogues of entropy. In *RANDOM-APPROX*, pages 200–215, 2003.
- [4] Eli Biham and Adi Shamir. Differential fault analysis of secret key cryptosystems. In Burton S. Kaliski Jr., editor, *CRYPTO'97*, volume 1294 of *LNCS*, pages 513–525, Santa Barbara, CA, USA, August 17–21, 1997. Springer-Verlag, Berlin, Germany.
- [5] Dan Boneh, Richard A. DeMillo, and Richard J. Lipton. On the importance of checking cryptographic protocols for faults (extended abstract). In Walter Fumy, editor, *EUROCRYPT'97*, volume 1233 of *LNCS*, pages 37–51, Konstanz, Germany, May 11–15, 1997. Springer-Verlag, Berlin, Germany.
- [6] Ran Canetti, Yevgeniy Dodis, Shai Halevi, Eyal Kushilevitz, and Amit Sahai. Exposure-resilient functions and all-or-nothing transforms. In Bart Preneel, editor, *EUROCRYPT 2000*, volume 1807 of *LNCS*, pages 453–469, Bruges, Belgium, May 14–18, 2000. Springer-Verlag, Berlin, Germany.
- [7] Ran Canetti, Dror Eiger, Shafi Goldwasser, and Dah-Yoh Lim. How to protect yourself without perfect shredding. *LNCS*, pages 511–523. Springer-Verlag, Berlin, Germany, 2008.
- [8] David Cash, Yan Zong Ding, Yevgeniy Dodis, Wenke Lee, Richard J. Lipton, and Shabsi Walfish. Intrusion-resilient key exchange in the bounded retrieval model. In Salil P. Vadhan, editor, *TCC 2007*, volume 4392 of *LNCS*, pages 479–498, Amsterdam, The Netherlands, February 21–24, 2007. Springer-Verlag, Berlin, Germany.
- [9] Giovanni Di Crescenzo, Richard J. Lipton, and Shabsi Walfish. Perfectly secure password protocols in the bounded retrieval model. In Shai Halevi and Tal Rabin, editors, *TCC 2006*, volume 3876 of *LNCS*, pages 225–244, New York, NY, USA, March 4–7, 2006. Springer-Verlag, Berlin, Germany.

- [10] Yevgeniy Dodis, Amit Sahai, and Adam Smith. On perfect and adaptive security in exposure-resilient cryptography. In Birgit Pfizmann, editor, *EUROCRYPT 2001*, volume 2045 of *LNCS*, pages 301–324, Innsbruck, Austria, May 6–10, 2001. Springer-Verlag, Berlin, Germany.
- [11] Stefan Dziembowski. Intrusion-resilience via the bounded-storage model. In Shai Halevi and Tal Rabin, editors, *TCC 2006*, volume 3876 of *LNCS*, pages 207–224, New York, NY, USA, March 4–7, 2006. Springer-Verlag, Berlin, Germany.
- [12] Stefan Dziembowski and Krzysztof Pietrzak. Intrusion-resilient secret sharing. In *48th FOCS*, pages 227–237, Providence, USA, October 20–23, 2007. IEEE Computer Society Press.
- [13] Stefan Dziembowski and Krzysztof Pietrzak. Leakage-resilient cryptography. In *49th FOCS*, pages 293–302. IEEE Computer Society Press, 2008.
- [14] Stefan Dziembowski and Krzysztof Pietrzak. Leakage-resilient cryptography in the standard model. Cryptology ePrint Archive, Report 2008/240, 2008. <http://eprint.iacr.org>.
- [15] Karine Gandolfi, Christophe Mourtel, and Francis Olivier. Electromagnetic analysis: Concrete results. In Çetin Kaya Koç, David Naccache, and Christof Paar, editors, *CHES 2001*, volume 2162 of *LNCS*, pages 251–261, Paris, France, May 14–16, 2001. Springer-Verlag, Berlin, Germany.
- [16] Timothy Gowers. Decompositions, approximate structure, transference, and the hahn-banach theorem, 2008. Preprint.
- [17] Ben Green and Terence Tao. The primes contain arbitrarily long arithmetic progressions. *Annals of Math.*, 167:481–547, 2008.
- [18] J. Alex Halderman, Seth D. Schoen, Nadia Heninger, William Clarkson, William Paul, Joseph A. Calandrino, Ariel J. Feldman, Jacob Appelbaum, and Edward W. Felten. Lest we remember: Cold boot attacks on encryption keys. In *USENIX Security Symposium*, pages 45–60, 2008.
- [19] 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.
- [20] Yuval Ishai, Manoj Prabhakaran, Amit Sahai, and David Wagner. Private circuits II: Keeping secrets in tamperable circuits. In Serge Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 308–327, St. Petersburg, Russia, May 28 – June 1, 2006. Springer-Verlag, Berlin, Germany.
- [21] Yuval Ishai, Amit Sahai, and David Wagner. Private circuits: Securing hardware against probing attacks. In Dan Boneh, editor, *CRYPTO 2003*, volume 2729 of *LNCS*, pages 463–481, Santa Barbara, CA, USA, August 17–21, 2003. Springer-Verlag, Berlin, Germany.
- [22] Jesse Kamp and David Zuckerman. Deterministic extractors for bit-fixing sources and exposure-resilient cryptography. In *44th FOCS*, pages 92–101, Cambridge, Massachusetts, USA, October 11–14, 2003. IEEE Computer Society Press.
- [23] John Kelsey, Bruce Schneier, David Wagner, and Chris Hall. Side channel cryptanalysis of product ciphers. In Jean-Jacques Quisquater, Yves Deswarte, Catherine Meadows, and Dieter Gollmann, editors, *ESORICS’98*, volume 1485 of *LNCS*, pages 97–110, Louvain-la-Neuve, Belgium, September 16–18, 1998. Springer-Verlag, Berlin, Germany.
- [24] Paul C. Kocher. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In Neal Koblitz, editor, *CRYPTO’96*, volume 1109 of *LNCS*, pages 104–113, Santa Barbara, CA, USA, August 18–22, 1996. Springer-Verlag, Berlin, Germany.
- [25] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. In Michael J. Wiener, editor, *CRYPTO’99*, volume 1666 of *LNCS*, pages 388–397, Santa Barbara, CA, USA, August 15–19, 1999. Springer-Verlag, Berlin, Germany.
- [26] Ueli M. Maurer. A universal statistical test for random bit generators. In Alfred J. Menezes and Scott A. Vanstone, editors, *CRYPTO’90*, volume 537 of *LNCS*, pages 409–420, Santa Barbara, CA, USA, August 11–15, 1991. Springer-Verlag, Berlin, Germany.
- [27] Silvio Micali and Leonid Reyzin. Physically observable cryptography (extended abstract). In *TCC*, pages 278–296, 2004.
- [28] Silvio Micali and Leonid Reyzin. Physically observable cryptography (extended abstract). In *TCC*, pages 278–296, 2004.
- [29] John Von Neumann. Zur Theorie der Gesellschaftsspiele. *Mathematische Annalen*, 100(1):295–320, 1928.
- [30] European Network of Excellence (ECRYPT). The side channel cryptanalysis

lounge. http://www.crypto.ruhr-uni-bochum.de/en_sclounge.html. retrieved on 29.03.2008.

- [31] Christophe Petit, François-Xavier Standaert, Olivier Pereira, Tal Malkin, and Moti Yung. A block cipher based pseudo random number generator secure against side-channel key recovery. In Masayuki Abe and Virgil Gligor, editors, *ASIACCS 08*, pages 56–65, Tokyo, Japan, March 18–20, 2008. ACM Press.
- [32] Jean-Jacques Quisquater and David Samyde. Electromagnetic analysis (ema): Measures and countermeasures for smart cards. In *E-smart*, pages 200–210, 2001.
- [33] Jean-Jacques Quisquater and François Koene. Side channel attacks: State of the art, October 2002. [30].
- [34] Omer Reingold, Luca Trevisan, Madhur Tulsiani, and Salil P. Vadhan. Dense subsets of pseudorandom sets. In *49th FOCS*, pages 76–85. IEEE Computer Society Press, 2008.
- [35] Luca Trevisan. Guest column: additive combinatorics and theoretical computer science. *SIGACT News*, 40(2):50–66, 2009.