IACR Transactions on Cryptographic Hardware and Embedded Systems
ISSN 2569-2925, Vol. 2025, No. 2, pp. 604–646. DOI:10.46586/tches.v2025.i2.604-646

Information Theoretic Analysis of PUF-Based Tamper Protection

Georg Maringer[1,2] and Matthias Hiller[2]

[1] Technical University of Munich, Munich, Germany, georg.maringer@tum.de
[2] Fraunhofer AISEC, Garching, Germany,
{georg.johannes.maringer,matthias.hiller}@aisec.fraunhofer.de

Abstract. PUFs enable physical tamper protection for high-assurance devices without needing a continuous power supply that is active over the entire lifetime of the device. Several methods for PUF-based tamper protection have been proposed together with practical quantization and error correction schemes. In this work we take a step back from the implementation to analyze theoretical properties and limits. We apply zero leakage output quantization to existing quantization schemes and minimize the reconstruction error probability under zero leakage. We apply wiretap coding within a helper data algorithm to enable a reliable key reconstruction for the legitimate user while guaranteeing a selectable reconstruction complexity for an attacker, analogously to the security level for a cryptographic algorithm for the attacker models considered in this work. We present lower bounds on the achievable key rates depending on the attacker's capabilities in the asymptotic and finite blocklength regime to give fundamental security guarantees even if the attacker gets partial information about the PUF response and the helper data. Furthermore, we present converse bounds on the number of PUF cells. Our results show for example that for a practical scenario one needs at least 459 PUF cells using 3 bit quantization to achieve a security level of 128 bit.

Keywords: Physical Unclonable Functions · Tamper Protection · Error Correction · Wiretap Channel · Secret Sharing · Physical Layer Security

1 Introduction

Physical Unclonable Functions (PUFs) evaluate physical properties of devices to obtain unique identifiers of electronic devices and provide physical roots of trust for cryptographic keys. Furthermore, PUFs can serve as a foundation for tamper protection technology that facilitates to validate the physical integrity of an embedded system after its power-up. All approaches have in common that minuscule manufacturing variations within physical objects, or mostly electronic components, are evaluated to generate an internal device-unique output. While there are several works on the assessment of the entropy or randomness of PUFs [MGS13, WGP18, FWHP23] and the leakage through the published helper data necessary within the reconstruction phase, e.g. [DGSV15, DGV$^+$16], we are currently lacking a theoretical model to quantify the security in the light of a physical attacker who destroys parts of the PUF response to read out the remainder.

Going from silicon PUFs, e.g. SRAM, ring oscillator or arbiter PUF [HYKD14], to system-level PUFs facilitates to incorporate tamper protection capabilities to protect an entire embedded device with components that cannot resist advanced physical attacks on their own, such as processors, FPGAs, or external memories and their communication, or discrete components that are susceptible e.g. to side-channel attacks. This reduces the attack surface from several individual vulnerabilities, e.g., against laser or EM fault

Received: 2024-10-15 Accepted: 2024-12-15 Published: 2025-03-04

injection, EM or optical side channel analysis, and analysis of digital communication interfaces between components to the attack resistance of the surrounding barrier. We also consider it infeasible to perform power side-channels since the attacker can only access the power supply of the entire printed circuit board and has no direct access to the power supply of the chip or individual discrete components.

For the remainder of this work, we will refer to the PUF-based tamper protection foil proposed by Immler *et al.* [IOK$^+$18, ION$^+$19]. However, the generic results of this work can be adapted and applied to other PUF types as well. Examples are the coating PUF [TSS$^+$06a] and the polymer waveguide PUF [VWN$^+$16, GGV17]. The tamper protection is based on a foil that is wrapped around an entire Printed Circuit Board (PCB), or a cover that is placed on top and bottom. This foil consists of a mesh of electrodes, leading to a large number of capacitances that can be measured by a mixed-signal circuit from within the protected area. It evaluates the capacitive coupling between electrodes to derive the cryptographic key and to validate the physical integrity of the system, and performs run-time tamper detection during operation to protect the system.

One of the critical attack vectors, considered during the evaluation of hardware devices with security boxes [SI23], is that an attacker penetrates the foil with a small needle or drill and accesses internal signals. If the required drill diameter is sufficiently large, major changes occur in the capacitance measurements of a significant portion of the foil, leading to an incorrectly reconstructed PUF response during the reconstruction phase. Therefore, the secret cannot be uncovered by an attacker, as discussed, e.g., in [GXKF22]. As the foil's PUF values may also change over time due to noise, aging, and varying environmental conditions such as temperature or humidity, an error-correcting code is implemented in the system to compensate for those effects to ensure that the correct cryptographic key is derived with a probability $> 1 - 10^{-6}$ or even $> 1 - 10^{-9}$ so that the PUF does not have significant impact on the reliability of the overall system.

Our goal is to analyze the resulting wiretap channel [Wyn75, CK78] between the enrollment and reconstruction phase of the legitimate user as well as the reconstruction phase of the attacker from an information theoretical point of view. We establish lower bounds on the secrecy capacities of the resulting channels as well as finite blocklength achievability and converse bounds on the maximal achievable secrecy rate, making our results relevant in practice as they provide benchmarks for implementations by quantifying the distance of a practical implementation to the theoretical limit.

1.1 Related Works

For a survey on information and coding theoretic techniques covering enrollment and reconstruction without tamper protection see [GS20]. We also mention literature in the context of biometric secrecy as this field is closely related to PUFs. [GİSK19] for example code constructions for both biometric secrecy systems as well as PUFs are given. Achievable rate regions of biometric secrecy systems under security and privacy constraints are presented in [IW09]. Approaches to achieve biometric secrecy using Slepian-Wolf distributed source coding techniques are presented in [VDRY09, DKM$^+$07].

1.2 Main Results

The main results presented in this work are:

- Information theoretical channel model including zero leakage helper data generation for physical tampering with PUFs
- Asymptotic results for lower bounds on the channel capacity of the resulting PUF-channel under different attack scenarios

- Finite blocklength achievability and converse results on the number of required capacitances to achieve a predefined security level in two attack scenarios
- Proof that previously used helper data schemes do not achieve required security levels without leaking information about the secret via the helper data
- Quantitative results that demonstrate that a 128 bit security level is achievable with 1400 PUF cells for 18% and 36% erasure probability for digital and analog attacker, respectively
- Proof that an existing converse bound on finite blocklength wiretap codes cannot be tight for all channels

1.3 Outline

Section 2 gives a brief overview over related work. Section 3 introduces the notation used throughout this work and recaps known results in the field of information theory, in particular for finite blocklength that are used to proof the main results presented in this work. Furthermore, helper data algorithms with an emphasis on zero leakage helper data are recapped and their connection between secret sharing using common randomness is examined. Section 5 gives some background information on the foil PUF and introduces the resulting channel model. In Section 6, we obtain results on the secret key capacity of the HDA for digital and analog attacker. Section 7 investigates secret sharing using common randomness using one-way communication for finite blocklength. It serves as a foundation to analyze the HDA performance with respect to the required amount of capacitive PUF cells for the foil PUF presented in Section 8. In Section 9 we use the converse result on the necessary amount of PUF cells to show that a given implementation has to either leak via the helper data or be insecure by other means. Section 10 sums up the results and states open problems.

2 State of the Art

While some silicon PUFs such as the SRAM or arbiter PUF directly output digital information, other silicon PUFs, like the Ring-Oscillator or TERO PUF, and in particular non-silicon PUFs output analog, or finely quantized digital data. They all have in common, that they undergo a processing chain involving helper data and error correction, until a cryptographic key is output (see Section 3.4).

2.1 PUF-Based Tamper Protection

In the past, tamper protection was implemented through a continuously powered detection mechanism, e.g. in tamper-responsive envelopes and covers that wrap or cover the structure to be protected [IMJFC13, OI18]. Within the protective structure, a physical measure such as electrical resistance is measured and triggers an alarm as soon as the measured value exceeds a threshold to erase sensitive information, stored e.g. in battery-backed memory. While the device is only active during a fraction of the time, the protective measures need to be active for the entire life-time of the device, after it leaves a trusted manufacturing site. PUF-based tamper protection promises to increase sensitivity and to ease the handling during operation of the devices, as the device can be fully powered off. Thus PUFs, can contribute to an easy-to-handle and future-proof tamper protection technology.

Over the last 20 years, different measures were taken to combine PUFs and tamper protection. An early type is the coating PUF, where a protective coating is spread over an IC and evaluated from its inside to detect physical tampering when the coating is

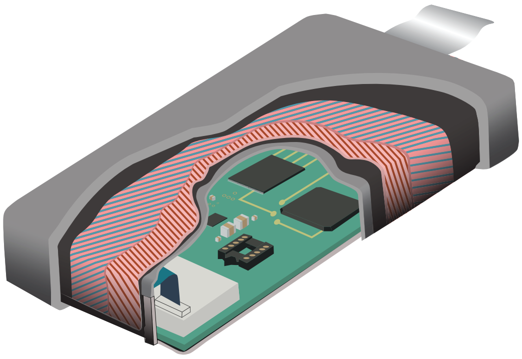

Figure 1: Sketch of PUF-based envelope [IOK+18] with two conducting electrode layers in red and blue, and two shielding layers in black

penetrated [SMKT06, TSS+06b]. In addition to electrical measurements, also optical approaches [EFK+12, VNK+15], mechanical properties [GS22] or propagation behavior of radio waves [STZP22] were proposed.

In this work, we use PUF-based tamper protection realized by foils and covers, as also proposed in [IOK+18, Imm19, ION+19, GOFK21] as reference. This scenario is depicted in Figure 1. The foil is wrapped around a PCB with mounted electronics components to be protected and consists of two structured electrode layers (blue and red) and two electrical shielding layers (black). The capacitive coupling between the electrode layers is measured to obtain the PUF response [OIHS18]. In addition, faster mechanisms for run-time tamper protection are included.

The capacitance values are subject to measurement noise, temperature and other environmental effects, as discussed based on measured values, e.g. in [GXKF22]. A broad range of deterministic effects can be compensated with linear or also higher order reference points of fits [OIHS18, GXKF22, RFB+23, GİK15], whereas Gaussian measurement noise remains in all cases. We will focus on this fundamental noise issue using a wiretap scenario in this work and refer to the compound case for including multiple environmental conditions [LKP09, BW13].

2.2 Quantization

Analog PUFs evaluated on embedded devices need to quantize the digitized PUF data into a finite alphabet that is input into the error correcting code (see Section 3.5). Typically, public helper data that references the distance and direction to the next interval border is stored to move a measured data point away from decision borders right into the middle of the quantization interval. So far, work on PUFs typically either uses equiprobable or equidistant quantization [IHKS16]. As shown in Section 3.6 it is possible though to use arbitrary input quantization while still generating helper data that is not leaking any information about the secret (zero-leakage helper data).

Equiprobable Quantization The range of possible output values of the PUF is split into d intervals with the same probability such that all indices as sampled with the same probability. This is favorable from a security point of view, as iid PUF values are mapped into uniform data in the finite alphabet. However, this comes with two downsides:

First, the common helper data generation bringing the expected value during reconstruction into the center of the interval leaks information about the secret, as the helper data pointers of different intervals have different distributions [IHKS16]. Later, we will introduce a method to obtain zero leakage helper data such that this problem is mitigated.

Second, the decision borders in the center are rather narrow, so that the values in this intervals are subject to a higher error rate. The wide intervals in the tails of the Gaussian distribution considered in our model also have a decreased sensitivity for physical tampering.

Equiprobable quantization can be applied if the requirement for uniformity is more substantial than the requirement for tamper detection sensitivity [GXKF22].

Equidistant Quantization In contrast, equidistant quantization samples the analog values by mapping them to intervals of the same size. This has the advantage that additive noise effects all values in the same way and error probabilities are constant with respect to the PUF values. Also, only a negligible leakage can be observed [IHKS16] through the aforementioned helper data pointers. With the later on introduced zero leakage helper data algorithms, this is less of a factor. As a downside, this comes at the expense of a heavy bias as the intervals differ in probability. This can be mitigated through variable-length encoding at the expense of a limited selection of code classes for the later ECC [IHL+18]. As shown in [BDH+10], also higher-dimensional structures can be used for embedding secret data, which adds more degrees of freedom.

2.3 Error Correction

Aside from the helper data used to perform better output quantization during reconstruction (denoted by W^n later in Section 3.4) additional helper data $\widetilde{W}$ is generated to link the PUF response to a codeword of an error correcting code. This can be done either by linear schemes that generate the link through linear dependencies such as syndromes or XORs or pointers that refer to parts of the PUF response [HKS20].

In any case, an ECC is used to reduce the key error probability e.g. down to 10^{-6} or 10^{-9} to generate reliable keys. This can be achieved with standard codes such as BCH, Reed-Solomon or Convolutional codes [MS77], or newer code classes such as limited magnitude codes [IU19] or Polar codes [CIW+17, GXKF22].

In addition, wiretap codes can be used either for leakage prevention [HO17] or attack prevention [GXKF22].

Over the last decades, several schemes have been proposed and implemented to address the design of the error correcting codes (ECCs) and helper data, see e.g. [DGSV15, HKS20]. In the following, we focus on the generation of input and output quantizers as well as fundamental theoretical limits on the amount of required PUF cells. Designers can then benchmark their implementations against the fundamental limits. The construction of a practical wiretap code and considerations for the leakage of the helper data connecting the PUF responses and the coding scheme are therefore out of the scope of this work.

3 Preliminaries

3.1 Notation

We denote scalars by lowercase letters and vectors by lowercase bold letters. Matrices are denoted by uppercase bold letters. We denote the i-th column of the matrix $\mathbf{A}$ by $\mathbf{a}_i$.

We denote random variables (RVs) by uppercase letters and their realizations by lowercase letters, i.e. the realization of a RV X is denoted by x. Furthermore, we denote the probability mass function of a discrete RV X by $P_X(x)$, the probability density function of a continuous RV X by $f_X(x)$ and the cumulative distribution by $F_X(x)$ in both cases. We denote the expectation operator by $\mathbb{E}[\cdot]$. If three random variables X, Y, Z form a Markov chain, i.e. $P_{XYZ}(x, y, z) = P_X(x)P_{Y|X}(y|x)P_{Z|Y}(z|y)$, we write $X \multimap Y \multimap Z$.

We denote the Gaussian distribution with mean μ and variance σ^2 by $\mathcal{N}(\mu, \sigma^2)$ and we write that a RV X is distributed according to a Gaussian by $X \sim \mathcal{N}(\mu, \sigma^2)$.

We denote the Gaussian Q-function by

$$Q(x) := \int_x^\infty \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right) \mathrm{d}z$$

and the complementary error function by

$$\operatorname{erfc}(x) := \frac{2}{\sqrt{\pi}} \int_x^\infty \exp(-z^2)\, \mathrm{d}z \ \ .$$

We define for a functions $f(n), g(n)$ the Landau symbols

$$f(n) \in o(g(n)) \Leftrightarrow \lim_{n\to\infty} \left|\frac{f(n)}{g(n)}\right| = 0$$

and

$$f(n) \in \mathcal{O}(g(n)) \Leftrightarrow \limsup_{n\to\infty} \left|\frac{f(n)}{g(n)}\right| < \infty \ \ .$$

We denote sets by caligraphic letters, e.g., a set $\mathcal{S}$ and its cardinality by $|\mathcal{S}|$.

To keep the paper self-contained, we provide selected basics in information and coding theory used in the following sections in Appendix A.

3.2 The Wiretap Channel

In [Wyn75] Wyner introduced a channel model, in the following referred to as the Degraded Wiretap Channel (DWTC) (see also [BB11]). The channel has one input, in the following denoted by the random variable A, and two outputs B and E accessed by Bob and Eve, respectively. This is illustrated in the upper part of Fig. 2. The goal of Alice is to use this channel to reliably transmit a message m to Bob, i.e. $m = \hat{m}$ with high probability while keeping any information about the message secret from an eavesdropper called Eve. For the degraded case it is obvious that the channel from Alice to Bob has higher channel capacity compared to the one from Alice to Eve. The joint distribution of the channel input and its output is given as $P_{ABE}(a, b, e) = P_A(a)P_{B|A}(b|a)P_{E|B}(e|b)$. The same does not necessarily hold for the more general wiretap channel (WTC) studied by Csiszar and Körner in [CK78] and shown in the lower part in Fig. 2. Here, the eavesdropper's channel output E is not a degraded version of B rather it is directly generated from A through a noisy channel, i.e. $P_{ABE}(a, b, e) = P_A(a)P_{B|A}(b|a)P_{E|A}(e|a)$ in this case.

The next definition is a slightly adapted version of the definition of secrecy codes in [YSP19].

Definition 1. An $(|\mathcal{M}|, \varepsilon, \delta)$ secrecy coding strategy (also referred to as a wiretap code) for a (degraded) wiretap channel $(\mathcal{A}, P_{B,E|A}, \mathcal{B} \times \mathcal{E})$ consists of

- a set of possible messages $\mathcal{M} := \{1, \ldots, |\mathcal{M}|\}$ from which a message $M = m$ is selected,
- a randomized encoder that generates a codeword $A(m)$ for $m \in \mathcal{M}$ according to a pdf $P_{A|M=m}$ and
- a decoder $Dec : \mathcal{B} \to \mathcal{M}$ that assigns an estimate $\hat{M}$ to each received signal $B \in \mathcal{B}$.

Encoder and decoder satisfy the average error probability (averaging performed by uniformly sampling the input message and over the randomness induced by the stochastic encoder)

$$Pr(Dec(B) \neq M) \leq \varepsilon$$

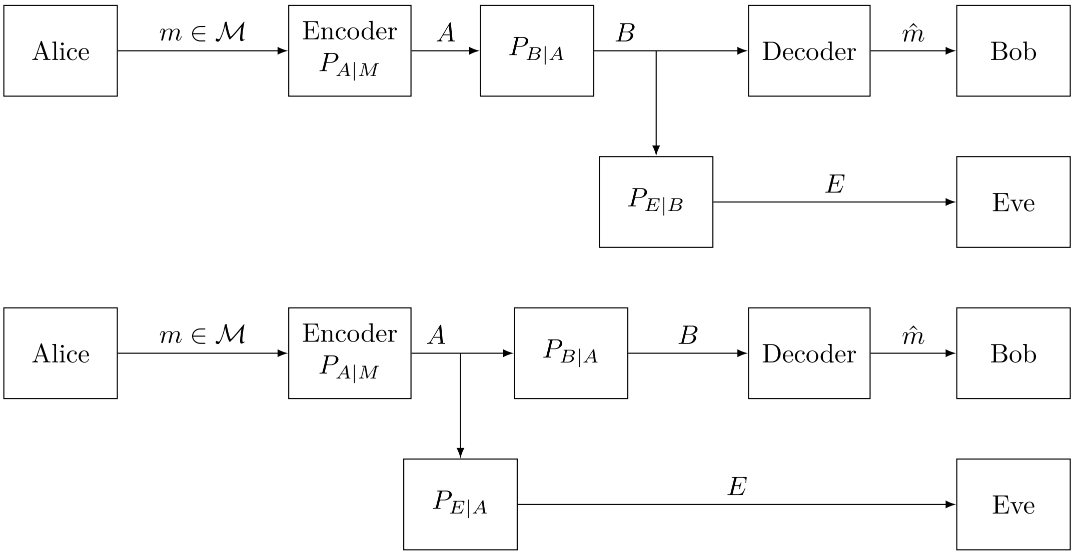

Figure 2: Degraded wiretap channel (top), wiretap channel (bottom)

where for B it holds that B is distributed according to $P_{B|M}(b|m) = \sum_{a\in\mathcal{A}} P_{B|A}(b|a)P_{A|M}(a|m)$. We distinguish average secrecy and maximum secrecy in the following way. For average secrecy with security parameter δ we require that

$$d(P_{ME}, P_M^{unif} P_E) \leq \delta \ . \tag{1}$$

where P_M^{unif} denotes the uniform distribution over the space of possible messages. In contrast for maximum secrecy it has to hold that

$$\max_{m\in\mathcal{M}} d(P_{E|M=m}, Q_E) \leq \delta \ , \tag{2}$$

where Q_E is the marginal distribution of E if uniformly distributed messages are transmitted over the channel. If we make the number of channel uses specific, we call an $(|\mathcal{M}|, \varepsilon, \delta)_{avg}$ average secrecy code for the channel $P_{B^nE^n|A^n}$ an $(n, |\mathcal{M}|, \varepsilon, \delta)_{avg}$ secrecy code. Codes for maximum secrecy are denoted by $(n, |\mathcal{M}|, \varepsilon, \delta)_{max}$. In the following we frequently omit specifying whether we are interested in the average or maximum secrecy setting. The definitions work analogously for both in those cases and when it is not clear from context we specify average or maximum secrecy in the index. We define the maximal achievable secrecy rate by

$$R^*(n, \varepsilon, \delta) := \max \left\{ \frac{\log(|\mathcal{M}|)}{n} : \exists (n, |\mathcal{M}|, \varepsilon, \delta) \text{ secrecy code} \right\} \ . \tag{3}$$

It makes intuitively sense that the security conditions in equations (1) and (2) make it hard for an attacker to obtain information about the message m. The following Theorem quantifies this statement for average secrecy.

Theorem 1 ([YSP16] Thm. 8)**.** *Let the output of an arbitrary list decoder $\mathcal{L}$ given Eve's observation E be denoted by $\mathcal{L}(E)$. Let δ be the secrecy parameter of the implemented secrecy code for the respective wiretap channel. Then the probability that the transmitted codeword is not in the output list of Eve's list decoder having listsize L is lower bounded by*

$$P_{ME}(M \notin \mathcal{L}(E)) \geq 1 - \delta - \frac{L}{|\mathcal{M}|} \ . \tag{4}$$

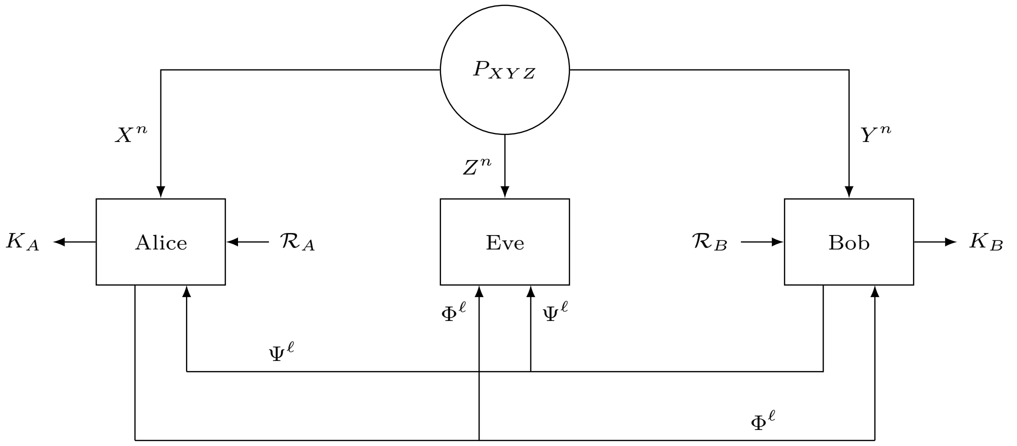

Figure 3: Secret Sharing using Common Randomness

Definition 2. The secrecy capacity C_S of a wiretap channel is defined by

$$C_S := \frac{1}{n} \limsup_{n\to\infty} R^*(n, \varepsilon, \delta) \tag{5}$$

for arbitrarily small values $\varepsilon > 0$ and $\delta > 0$.

The secrecy capacity for both channels is well known and given in the following Theorems.

Remark 1. Notice that we did not distinguish the secrecy capacity for average and maximal secrecy because their value is the same. However for fixed n, ε, δ, the values $R^*_{avg}(n, \varepsilon, \delta)$ and $R^*_{max}(n, \varepsilon, \delta)$ can be different.

Theorem 2 ([Wyn75]). *For the degraded wiretap channel with input A and outputs B and E for legitimate and eavesdropper, respectively, the secrecy capacity is equal to*

$$C_S = \max_{P_A} I(A; B) - I(A; E) \ . \tag{6}$$

Theorem 3 ([CK78]). *For the wiretap channel with input A and outputs B and E for legitimate and eavesdropper, respectively, the secrecy capacity is equal to*

$$C_S = \max_{P_{V,A}} I(V; B) - I(V; E) \ , \tag{7}$$

where V serves as an auxiliary random variable and it holds that $V \multimap A \multimap (B, E)$.

Remark 2. Notice that in both cases the capacity does not depend on ε and δ. This changes for the task of determining $R^*(n, \varepsilon, \delta)$ for finite n. We introduce bounds on $R^*_{avg}(n, \varepsilon, \delta)$ and $R^*_{max}(n, \varepsilon, \delta)$ for finite n later in Section 4.1.

3.3 Secret Sharing using Common Randomness

Secret sharing using common randomness has been investigated by Maurer in [Mau93] and by Ahlswede and Csiszàr in [AC93]. The problem is graphically illustrated in Fig. 3. In this section we explain the known results which will later be used in Section 6. In the following we describe the problem of deriving a secret key that is shared between two terminals, in the following called Alice and Bob, when the terminals have access to common randomness and are able to send messages to each other over a public channel.

The source of common randomness is specified by a joint probability mass function P_{XYZ}. We denote the respective random variables specifying its outputs by X, Y, Z. This source is iid sampled n times and we denote the random variables specifying the output of this process by (X^n, Y^n, Z^n). The first terminal, in the following denoted by Alice, receives the sequence $X^n = (X_1, \dots, X_n)$, while the second terminal (Bob) gets the sequence $Y^n = (Y_1, \dots, Y_n)$. The third terminal (Eve), which is an adversary trying to obtain knowledge about the secret key that Alice and Bob shall agree on, is provided with the sequence $Z^n = (Z_1, \dots, Z_n)$. The probability mass function P_{XYZ} is publicly known, in particular by Alice, Bob and Eve. The goal of Alice and Bob is to reliably agree on a secret key while leaving Eve oblivious about it. To achieve this goal, Alice and Bob send messages to each other over the public channel that depend on their apriori knowledge of P_{XYZ} and their respective shares X^n or Y^n. Subsequent messages may also depend on previously received messages over the public channel coming from the other terminal. Eve is able to eavesdrop those messages but is unable to alter them or to insert additional messages into the public channel. We denote the i-th message sent from Alice to Bob by Φ_i and the i-th message sent from Bob to Alice by Ψ_i. Furthermore, we define $\Phi^i := (\Phi_1, \dots, \Phi_i)$ and $\Psi^i := (\Psi_1, \dots, \Psi_i)$. After Alice and Bob are finished with exchanging messages over the public channel, say after ℓ steps, Alice computes a key K_A and Bob computes a key K_B that are both within the same keyspace denoted by $\mathcal{K}$. For the case that both keys are equal we simply denote the key by K.

As for transmitting data securely over a wiretap channel, it is essential for Alice and Bob to have access to local randomness to randomize the encoding function for the messages sent over the public channel. Hence, we assume that Alice and Bob have access to local sources of randomness $\mathcal{R}_A$ and $\mathcal{R}_B$, respectively. In order to generate the messages to be transmitted over the public channel they make use of those such that it holds that

$$\Phi_1 = \Phi_1(\mathcal{R}_A, x^n), \quad \Psi_1 = \Psi_1(\mathcal{R}_B, y^n) \tag{8}$$

$$\Phi_i = \Phi_i(\mathcal{R}_A, x^n, \Psi^{i-1}), \quad \Psi_i = \Psi_i(\mathcal{R}_B, y^n, \Phi^{i-1}) \ , \tag{9}$$

where $\mathcal{R}_A$ and $\mathcal{R}_B$ are independent from the jointly distributed random variables X^n and Y^n corresponding the source of common randomness.

We next formalize the secret key rate which is the figure of merit that we would like to maximize for this problem.

Definition 3. A secret key rate R is called **achievable** if for every $\varepsilon > 0$ and sufficiently large n there exists a secret key agreement scheme such that

1. $Pr(K_A \neq K_B) < \varepsilon$
2. $I(Z^n, \Phi^\ell, \Psi^\ell; K) < \varepsilon$
3. $H(K) > R - \varepsilon$
4. $\log_2(|\mathcal{K}|) < H(K) + \varepsilon$.

We next give some interpretation to the properties that an achievable secret key rate has according to Definition 3.

The first property basically states that for sufficiently large n the probability that the key at the Alice terminal is unequal to the key at Bob's terminal is arbitrarily small. The second property states that no information can be deduced from the messages shared over the public channel and the source component Z^n about the key K. We recall at this point that random variables are stochastically independent if and only if their mutual information is zero and the second property says that we are able to approach this arbitrarily closely. The third property states that the entropy of K is basically at least R because ε is arbitrarily small. The fourth property states that the key is almost uniform over the keyspace $\mathcal{K}$.

The natural question of finding the maximal achievable secret key rate, in the following referred to as the *secret key capacity* as a function of P_{XYZ} has been answered in [AC93] and [Mau93].

Theorem 4. *The* ***secret key capacity*** $\widetilde{C}_S$ *denotes the maximal achievable secret key rate for a source of common randomness* P_{XYZ} *and is bounded by*

$$I(X;Y) - I(X;Z) \leq \widetilde{C}_S \leq I(X;Y|Z) \ . \tag{10}$$

Furthermore, the secret key capacity is achievable even if we only allow a single transmission over the public channel from Alice to Bob or from Bob to Alice.

Corollary 1 ([BB11], Corollary 4.1)**.** *If* $X \multimap Y \multimap Z$ *it holds that*

$$\widetilde{C}_S = I(X;Y) - I(X;Z) = I(X;Y|Z) \tag{11}$$

and hence upper and lower bound in (10) *are matching.*

Remark 3. Notice that the secret key capacity C_S is very similar to the secrecy capacity of a degraded wiretap channel $P_{XYZ}(x,y,z) = P_X(x)P_{Y|X}(y|x)P_{Z|Y}(z|y)$ (see equation (6)) except for the fact that a maximization over the distribution P_X is omitted.

Before we give a proof sketch for the achievability part of Theorem 4 we examine some special cases for P_{XYZ}. Let us assume that either $Z^n = X^n$ or $Z^n = Y^n$ holds. In this case the secret key capacity is zero. This is intuitively appealing as Eve is able to observe all communication over the public channel and one legitimate party has the same information from the source of common randomness as Eve. On the other hand if $P_{XYZ}(x,y,z) = P_{XY}(x,y)P_Z(z)$, observing Z^n gives no information about the shares X^n or Y^n. Hence, Z^n provides no useful information to Eve at all which is reflected by the fact that $I(X;Z) = 0$ in that case.

Remark 4. In [AC93] it has first been discussed how to perform secret sharing with common randomness if the eavesdropper only has access to the messages transmitted over the public channel, i.e. the source was of the form P_{XY} and only later to examine the more general case (including Z). This is essentially equivalent to the case that Z is independent of X, Y. In this work, we chose the approach of directly introducing the model in the more general setting (including Z) as was also done in [Mau93].

Proof sketch. In the following we sketch how the secret key rate given in Theorem 4 can be achieved using secrecy codes designed for secure data transmission over a (potentially degraded) wiretap channel. Furthermore, this construction only requires the transmission of a single block of length n from Alice to Bob over the public channel. We follow the structure of the proof given in [BB11, Chapter 4.2.1].

Let the secret key be encoded into a block of n symbols over $\mathcal{X}$, labelled by u^n. Alice would like to securely communicate the i-th symbol $u_i \in \mathcal{X}$ to Bob. The choice of this symbol is stochastically independent of the common randomness outputs X^n, Y^n, Z^n. She computes $u_i + x_i$ and sends the result over the public channel, where the addition is taken mod $|\mathcal{X}|$. Bob receives his i-th dedicated symbol from the source of common randomness y_i and $u_i + x_i$ from the public channel. Eve receives z_i from the common randomness source and is able to eavesdrop $u_i + x_i$. This scenario can be interpreted as a wiretap channel. The channel's input is U, while the legitimate user's channel output is formed by the tuple $(U + X, Y)$ and the eavesdropper's output is formed by $(U + X, Z)$.

Assume that the sequence all possible sequences u^n are codewords of a wiretap code. It is possible to construct such a code by sampling the elements of all codewords from a single

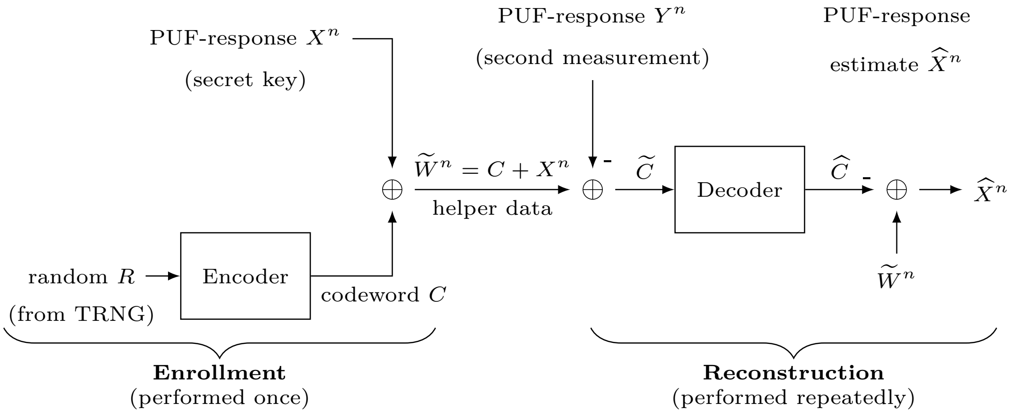

Figure 4: Simplified schematic of a key generation scheme based on a PUF.

distribution P_U that can be chosen arbitrarily. According to the standard achievability proofs for secrecy codes of wiretap channels (see for instance [BB11, Chapter 3.4.1]) From Theorem 3 we know that a secrecy code for this channel with rate

$$R^* := I(U; Y, U + X) - I(U; Z, U + X) = H(U|Z, U + X) - H(U|Y, U + X) \tag{12}$$

is achievable by choosing the auxiliary random variable $V = U$. In case $X \multimap\!\!- Y \multimap\!\!- Z$ it follows that $U \multimap\!\!- (U + X, Y) \multimap\!\!- (U + X, Z)$ and by Theorem 2 that R^* is achievable.

For the choice of U being uniformly distributed over $\mathcal{X}$ and using properties of the one-time pad one is able to show that

$$R^* = I(X; Y) - I(X; Z) \ . \tag{13}$$

Hence, we have constructed a secret sharing scheme using common randomness from the source P_{XYZ} achieving the rate $I(X; Y) - I(X; Z)$. ■

The reason we provide this proof is that the methodology to perform secret sharing using common randomness using secrecy codes for wiretap channels naturally carries over to the finite blocklength regime. We come back to this point in Section 7.

3.4 Helper Data Algorithm

In the following we present a widely used method to integrate error correction capabilities into the generation of a cryptographic key using PUFs. We refer to this methodology as the helper data algorithm (HDA) [DRS04a]. A block diagram for the helper data algorithm is shown in Fig. 4.

The PUF response X^n, in this example the content of multiple SRAM cells after power up, is measured during the manufacturing process of the device, which is referred to as *enrollment*. Furthermore, during the enrollment a random number R is sampled from a True Random Number Generator (TRNG) which is used to select a codeword C at random from an ECC and the helper data $\widetilde{W}^n$ is computed according to $\widetilde{W}^n := C + X^n$. This helper data is published, e.g., in an external storage on the embedded system. Notice that the amount of SRAM cells corresponds to the length of the code in this case. We remark that additions and subtractions in this section are usually performed within the finite field over which the ECC is defined.

The goal of the helper data algorithm is to obtain the PUF response measured during the enrollment at another time when the PUF is measured again. We call this process *reconstruction*. The PUF measurement during the reconstruction phase is denoted by the random variable Y^n. Using the helper data $\widetilde{W}^n$ we compute $\widetilde{C} := \widetilde{W}^n - Y^n$. Since Y^n does not have to be equal X^n even if the same device is used during the reconstruction phase we have that $\widetilde{C} = C + E^n$, where $E^n = X^n - Y^n$ denotes the error vector. If E^n is of sufficiently small weight, the codeword is correctly decoded. We denote the decoder's output by $\widehat{C}$. We then estimate the PUF response after applying error correction $\widehat{X}^n$ by computing $\widehat{X}^n = \widetilde{W}^n - \widehat{C}$.

The purpose of the helper data algorithm can be abstracted to the fact that it is not possible to perform coding over the channel from X^n to Y^n specified by the probability mass function $P_{Y^n|X^n}(y^n|x^n)$ as those values are the outputs of the PUF measurement without inherent structure. This channel is a function of the measurement noise and potential temperature dependence or aging effects. The helper data algorithm enables the integration of a structured ECC that can be chosen by the designer. This enables error correction within the reconstruction phase. Notice that the elements of the codeword C and the first estimate before decoding $\widetilde{C}$ are connected by virtually the same channel is C and X^n as well as $\widetilde{C}$ and Y^n differ only additively by the helper data $\widetilde{W}^n$. From the construction it also becomes obvious that the security level of the helper data scheme is upper bounded by the code dimension of the ECC in bits as this is simply the brute force complexity. The length of the code on the other hand determines the required length of the PUF-response and hence can be directly associated with the hardware complexity of the PUF, i.e. via the required number of SRAM cells for an SRAM PUF.

To remove any bias from the PUF-response, frequently the generated secret key is not equal to the PUF-response X^n but corresponds to the hash-value of X^n using a cryptographically secure hash-function. In general an approximate reconstruction of X^n is insufficient for key derivation because slight changes at the input of cryptographic functions, e.g. via the key, typically already lead to substantial changes at the function's output.

3.5 Helper Data Algorithm for Analog PUFs

In contrast to a PUF outputting digital values like in the SRAM PUF, the output of analog PUFs cannot be directly fed into the helper data algorithm introduced in Section 3.4. Therefore, during the enrollment phase the analog PUF output X^n needs to be quantized by an input quantizer and $S^n = Q(X^n)$. Furthermore, additional helper data is generated from X^n for the quantization required in the reconstruction phase. We denote this function by g and the resulting quantization helper data by W^n, in particular $W^n = g(X^n)$. A block diagram illustrating the additional quantization and helper data generation steps is given in Fig. 5. Input quantization and helper data generation are performed elementwise. With slight inaccuracy in notation, we sometimes also write $Q(X)$ for the quantization of a single PUF value. $g(X)$ is treated the same way. We denote the codomain of Q by $\mathcal{S}^n$ and the codomain of g by $\mathcal{W}^n$. The quantization helper data W^n is published while S^n forms the key or is the preimage of the key via a cryptographically secure hash function and hence is kept secret. Like in the discrete case, the ECC is fixed and a random number R sampled from a TRNG to determine a codeword C. Similar to digital PUFs, the helper data algorithm computes and publishes $\widetilde{W}^n = C + S^n$.

During the reconstruction phase, the user measures the PUF again. We denote the outcome of this measurement by Y^n. From the quantization helper data W^n an output quantizer is derived which is then used to output an estimate $\widetilde{S}^n$ of S^n. The remaining steps are analogous to the discrete case. The decoder basically decodes the erroneous codeword $\widetilde{C} = \widetilde{W}^n - \widetilde{S}^n$. Then the decoder outputs $\widehat{C}$ and $\widehat{S}^n = \widetilde{W}^n - \widehat{C}$ is computed. The key will be correctly recovered if $S^n = \widehat{S}^n$.

Since the quantization helper data W^n is derived directly from the PUF measurement during the enrollment phase X^n and so is the secret S^n, it has to be assured that from the public W^n it is impossible to derive information about S^n. The next section therefore

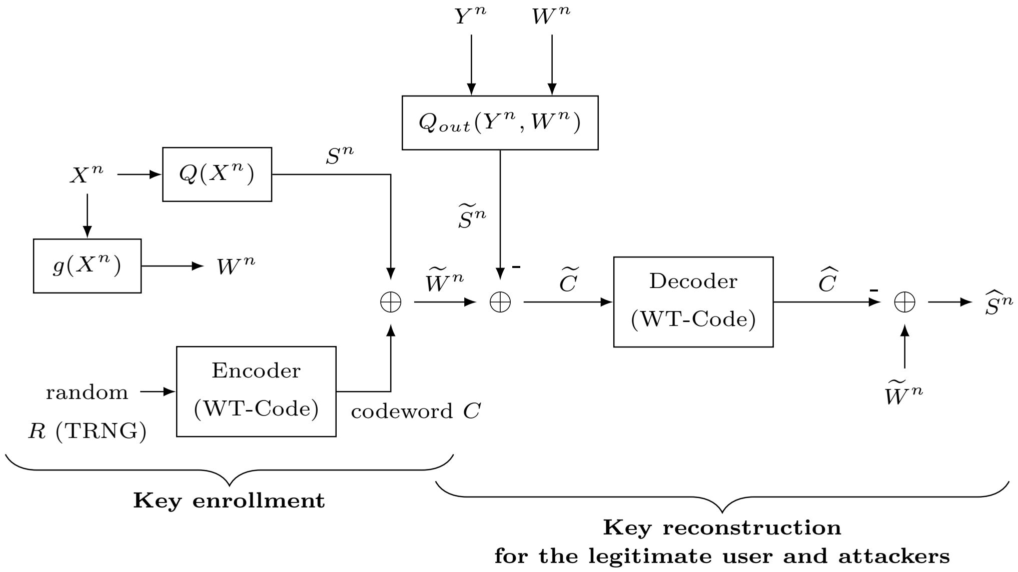

Figure 5: Key enrollment and key reconstruction for analog PUFs

deals with establishing quantization helper data that does not leak information about the secret S^n.

3.6 Zero Leakage Helper Data

Next, we give the definition of zero leakage for helper data algorithms. This is the notion that we are aiming at for the generation of the quantization helper data in analog PUFs.

Definition 4 ([dGSdVL16])**.** A helper data algorithm is defined to have *zero leakage* if the PUF response (after quantization) S^n and the quantization helper data W^n are stochastically independent, i.e.

$$P_{S^n|W^n}(s^n|w^n) = P_{S^n}(s^n),\ \forall s^n \in \mathcal{S}^n, w^n \in \mathcal{W}^n \ \ .$$

As in [dGSdVL16] we use a slightly stricter definition of zero leakage compared to the more standard definition in [VTO+10] to avoid pathological cases because we have continuous values for the helper data W^n.

Sufficient and necessary conditions for helper data featuring zero information leakage about the secret are given in [dGSdVL16]. In this work it was shown that it is possible to construct a zero leakage helper data scheme using a function g on the PUF response to generate the helper data scheme, having the following properties:

1. g is strictly monotonous function (and therefore an injective) function from each quantization interval to the domain of the helper data $\mathcal{W}$.

2. Any other function g^* generating the helper data cannot lead to a better reconstruction performance or does not have the zero leakage property.

The main result that we are using is given below:

Theorem 5 (Thm. 4.8 [dGSdVL16])**.** *Let g be monotonously increasing on each quantization interval A_t, with $g(A_0) = \ldots = g(A_{N-1}) = \mathcal{W}$, where N denotes the number of* *quantization levels. Let x_t and x_u be from different intervals with $g(x_t) = g(x_u)$. Then in order to satisfy zero leakage the following condition is sufficient and necessary:*

$$\frac{F_X(x_t) - F_X(q_t)}{p_t} = \frac{F_X(x_u) - F_X(q_u)}{p_u} \quad , \tag{14}$$

where q_t denotes the left border of the interval A_t and p_t denotes the probability that X is sampled to be in A_t, i.e. $p_t = Pr(X \in A_t)$. Analogous statements hold for q_u, A_u and p_u.

We refer to points x_t, x_u from different quantization intervals leading to the same helper data as *sibling points*. Theorem 5 also leads to a natural way of computing helper data via

$$w = g(x_{t,w}) := \frac{F_X(x_{t,w}) - F_X(q_t)}{p_t} \quad , \tag{15}$$

where $x_{t,w}$ denotes the point within the interval A_t leading to a helper data value of w. This is not the only optimal way to define the helper data but there is no way that leads to better performance during reconstruction while keeping the zero leakage property. Therefore, we take this approach for computing helper data throughout this work.

Lemma 1 ([SAS17] Lem. 1)**.** *The distribution of the helper data defined in equation* (15) *is the uniform distribution over the interval* $[0, 1]$*, i.e.*

$$f_W(w) = \begin{cases} 1 & \textit{for } w \in [0,1] \\ 0 & \textit{otherwise} \ . \end{cases} \tag{16}$$

Note that this construction facilitates to use the previously discussed equiprobable and equidistant input quantizations into zero leakage helper data algorithms.

In order to obtain the estimate $\widetilde{S}$ from the measurement during reconstruction Y the helper data is used to generate another quantizer. Its interval borders depend on W. For an equiprobable quantization, meaning that the input quantizer is formed such that S is uniformly distributed over its range, the construction of the output quantizer is also given in [dGSdVL16].

The more general case for arbitrary input distributions has been investigated in [SAS17]. The authors present an output quantizer for zero leakage helper data that we are also using in the following. The computation of the interval borders of this quantizer is given in Theorem 6. We refer to Remark 6 for a more specific statement on what is meant by $p_{t-1} \not\ll p_t$.

Theorem 6 ([SAS17] Thm. 1)**.** *Let the values $\tau_0, \ldots, \tau_N$ denote the interval borders of the output quantizer used to obtain $\widetilde{S}$ from Y and let $p_{t-1} \not\ll p_t$. Let $\tau_0 = -\infty$ and $\tau_N = \infty$. Let $g_t^{-1}(w)$ be the unique value x in quantization interval t such that $g(x) = w$. Then choosing τ_t iteratively starting from following equation* (17) *gives the best reconstruction estimate for zero leakage helper data.*

$$\tau_s = \frac{\ln\left(\frac{p_{t-1}}{p_t}\right)}{g_t^{-1}(w) - g_{t-1}^{-1}(w)} \sigma_N^2 + \frac{g_{t-1}^{-1}(w) + g_t^{-1}(w)}{2} \tag{17}$$

Remark 5. Notice that the choice for the output intervals given in Theorem 6 is consistent with Theorem 5.2 in [dGSdVL16] for uniform S.

Remark 6. If $p_s \ll p_{s-1}$ holds the symbol $s \in \mathcal{S}$ may be suboptimal irrespective of the channel output. This happens because the a-priori probability p_s of the symbol s is so large that for the equality point τ_s splitting the decision regions between the symbols

$s-1$ and s it holds that $\tau_s < \tau_{s-1}$. If this happens the output quantizer has only $|\mathcal{S}|-1$ symbols and we compute

$$\tau^* = \frac{\ln\left(\frac{p_{s-2}}{p_s}\right)}{g_s^{-1}(w) - g_{s-1}^{-1}(w)}\sigma_N^2 + \frac{g_{s-1}^{-1}(w) + g_s^{-1}(w)}{2} \quad . \tag{18}$$

In case $\tau^* < \tau_{s-2}$ we repeat the procedure. In this case the quantizer has only $|\mathcal{S}|-2$ symbols.

The computation of the output intervals can either be done during the reconstruction phase or within the enrolment phase. In the latter case the helper data is not a single number $w \in [0,1)$ but rather the set of all interval borders τ_s. This means that one can trade computational complexity during the reconstruction phase against storage consumption within the device, where the helper data needs to be stored.

Remark 7. Observe that the reconstruction quantizer depends on the distribution of S. This distribution in turn depends on the input quantizer and since $\widetilde{S}$ depends on the reconstruction quantizer we have the peculiar case that the conditional pmf $P_{\widetilde{S}|S,W}$ in general depends on the input distribution P_S.

3.7 Relation between Secret Sharing with Common Randomness and PUFs

Recalling the secret sharing problem using common randomness presented in Section 3.3 and comparing it to the problem of reconstructing the value of a PUF during the reconstruction phase that has previously been measured throughout the enrollment phase, we observe that the problems are almost equivalent. This equivalence is outlined in the following.

The PUF responses during enrollment X^n and reconstruction Y^n can be interpreted as n samples from a source of common randomness specified by a joint distribution P_{XY}. Notice that in this model there is no adversarial output Z for the moment. The helper data for reconstruction $\widetilde{W}^n$ which is published by the helper data algorithm can be interpreted as data being sent over the public channel by the terminal having the enrollment data X^n (Alice in the secret sharing scenario). The reconstruction measurement data Y^n (Bob's share of the source of common randomness) is then used together with the message $\widetilde{W}^n$ received from the public channel. For a random code with codewords that are iid sampled from the uniform distribution over the range of the RV X, the achievability proof sketch of Theorem 4 shows that the helper data scheme achieves the secret key capacity for the secret sharing problem using the resulting correlated source emanating from the PUF reconstruction problem. Notice that it is not possible to perform forward and backward transmissions over the public channel in this equivalence between secret sharing using common randomness and the PUF reconstruction problem. In the asymptotic setting it has been shown that one-way communication suffices to achieve the secret key capacity. In the finite blocklength regime this is unclear or even suggested not to hold (see Section 3.3). The construction achieving the secret key rate in Theorem 8 requires two-way communication over the public channel. Hence, it is unclear whether it is possible to generate a (possibly different) helper data scheme such that the rate in Equation (28) can be achieved for the coderate of the ECC. In fact, it is not obvious whether the helper data algorithm is optimal in maximizing the achievable rate if we restrict ourselves to one-way communication over the public channel. This question however is out of scope of this work.

4 Finite Blocklength Information Theory

In the previous sections we summarized results that make claims in the asymptotic setting as n goes to infinity. However, in practice we have to limit ourselves to some finite value for the blocklength n. For data transmission over point to point channels, natural questions to be asked are which rate R can be achieved for a given block length n and fixed block error probability P_e or which P_e can be achieved for fixed R and n.

4.1 Degraded Wiretap and Wiretap Channels

The finite blocklength behaviour of degraded wiretap and wiretap channels has already been investigated in [YSP16, YSP19]. In the finite blocklength regime the achievable rates depend on the concrete values for block error probability of the legitimate user ε and the security parameter δ. In contrast, similar to DMCs those parameters can be made arbitrarily small in the asymptotic regime as long as the rate is below the secrecy capacity (see Section 3.2).

The theorems dealing with the asymptotic behaviour (Theorem 2 and Theorem 3) merely provide the information that there exists a secrecy code of message cardinality

$$|\mathcal{M}| = 2^{nC_S + o(n)} \quad . \tag{19}$$

C_S is only an approximation of $R^*(n, \varepsilon, \delta)$ and estimating $R^*(n, \varepsilon, \delta)$ by C_S is only reasonable for very large n. For small and moderate n it is essential to further analyze the $o(n)$ term in Equation (19). The following theorem established in [YSP19] gives more precise upper and lower bounds on the exponent in Equation (19). The error term for these bounds is in the order of $\mathcal{O}(\log(n)/n)$.

Theorem 7 ([YSP19] Thm. 13)**.** *For a discrete memoryless (degraded or general) wiretap channel $P_{BE|A}$ with secrecy capacity C_S and for $\varepsilon + \delta < 1$ it holds that*

$$R^*_{max}(n, \varepsilon, \delta) \geq C_S - \sqrt{\frac{V_1}{n}} Q^{-1}(\varepsilon) - \sqrt{\frac{V_2}{n}} Q^{-1}(\delta) + \mathcal{O}\left(\frac{\log(n)}{n}\right) \tag{20}$$

and

$$R^*_{avg}(n, \varepsilon, \delta) \leq C_S - \sqrt{\frac{V_c}{n}} Q^{-1}(\varepsilon + \delta) + \mathcal{O}\left(\frac{\log(n)}{n}\right) \quad , \tag{21}$$

where

$$V_1 := \sum_{a \in \mathcal{A}} P_A(a) \left(\sum_{b \in \mathcal{B}} P_{B|A}(b|a) \log_2^2 \left(\frac{P_{B|A}(b|a)}{P_B(b)} \right) - D(P_{B|A=a} || P_B)^2 \right) \quad , \tag{22}$$

$$V_2 := \sum_{a \in \mathcal{A}} P_A(a) \left(\sum_{e \in \mathcal{E}} P_{E|A}(e|a) \log_2^2 \left(\frac{P_{E|A}(e|a)}{P_E(e)} \right) - D(P_{E|A=a} || P_E)^2 \right) \quad , \tag{23}$$

and

$$V_c := \sum_{a \in \mathcal{A}} P_A(a) \Bigg(\sum_{b \in \mathcal{B}, e \in \mathcal{E}} P_{BE|A}(b, e|a) \log^2 \left(\frac{P_{BE|A}(b, e|a)}{P_{E|A}(e|a) P_{B|E}(b|e)} \right) \tag{24}$$
$$- D(P_{BE|A=a} || P_{B|E} P_{E|A=a})^2 \Bigg) \quad .$$

Q^{-1} denotes the inverse of $Q(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right) \mathrm{d}z$.

To this point, it is unclear to us whether the construction using wiretap coding is optimal in terms of maximizing the secret key rate in the finite blocklength regime. To the best of our knowledge it is even unknown whether it is possible to achieve the secret key rate in Theorem 8 using one-way communication over the public channel or not. For the interested reader, we refer to the discussion on the necessity of two-way communication to achieve the secret key rate determined by Theorem 8 in [HTW16, Section VII].

4.2 Secret Sharing using Common Randomness

In this section we provide results for secret sharing with common randomness in the finite blocklength regime. This problem has been studied in [HTW16]. The authors established the secret key rate up to an error term in the order of $\mathcal{O}(\log(n)/n)$.

For the finite length setting we use the secret key definition given in [HTW16] which we present below. Notice that the security condition matches with the one of secrecy codes for wiretap channels given in Definition 1.

Definition 5. A secret sharing protocol is defined to achieve ε reliability and with average secrecy parameter δ if the probability that the two legitimate partners Alice and Bob fail to agree on the same key with probability less than ε and it holds that

$$d(P_{K,\Phi^\ell,\Psi^\ell,Z^n}, P_K^{unif} P_{\Phi^\ell,\Psi^\ell,Z^n}) \leq \delta \ . \tag{25}$$

where K denotes the secret key, Z^n denotes the share of the source of common randomness that the eavesdropper obtains and Φ^ℓ, Ψ^ℓ denote the communication over the public channel. For maximum secrecy the security condition is changed to

$$\max_{k\in\mathcal{K}} d(P_{Z^n|K=k}, Q_{Z^n,\Phi^\ell,\Psi^\ell}) \leq \delta \ , \tag{26}$$

where $Q_{Z^n,\Phi^\ell,\Psi^\ell}$ denotes the marginal distribution of $(Z^n, \Phi^\ell, \Psi^\ell)$ if uniformly distributed keys are considered. We define the maximal achievable secret key rate with blocklength n by

$$\widetilde{R}^*(n,\varepsilon,\delta) := \max\left\{\frac{\log(|\mathcal{K}|)}{n} : \exists (n, |\mathcal{K}|, \varepsilon, \delta) \text{ secret sharing protocol}\right\} \tag{27}$$

and specify by indices whether we mean the average or the maximum secrecy definition.

Theorem 8 ([HTW16],Thm.15)**.** *For every $\varepsilon, \delta > 0$ such that $\varepsilon + \delta < 1$ and iid (X^n, Y^n, Z^n) sampled according to joint pmf P_{XYZ} such that $X \multimap Y \multimap Z$, the maximal secret key rate for n, ε, δ is given by*

$$\widetilde{R}^*_{avg}(n,\varepsilon,\delta) = C_S - \sqrt{\frac{V_c'}{n}} Q^{-1}(\varepsilon+\delta) + \mathcal{O}\left(\frac{\log(n)}{n}\right) \ , \tag{28}$$

where

$$\begin{aligned} V_c' := \sum_{x\in\mathcal{X}} P_X(x) \Bigg(& \sum_{y\in\mathcal{Y}, z\in\mathcal{Z}} P_{YZ|X}(y,z|x) \log_2^2\left(\frac{P_{YZ|X}(y,z|x)}{P_{Z|X}(z|x)P_{Y|Z}(y|z)}\right) \\ & - D(P_{YZ|X=x}||P_{Y|Z}P_{Z|X=x})^2 \Bigg) \ . \end{aligned} \tag{29}$$

Examining equation (28), we observe that its structure is similar to the bounds on the maximal secrecy rate in Theorem 7. The first term is the asymptotic result, i.e. the secret key capacity defined by the source of common randomness. This value is independent of n, ε, δ and only depends on P_{XYZ}. Another term that depends on P_{XYZ} via the dispersion coefficient V_c but also on n, ε, δ is subtracted from this value. Notably, this term decreases in n with speed $1/\sqrt{n}$ and hence Theorem 8 is consistent with the asymptotic result presented in Theorem 4. Finally, there is an error term in the order of $\mathcal{O}(\log(n)/n)$ that decreases significantly faster than $1/\sqrt{n}$ such that ignoring it approximates reality reasonably well for moderate values of n.

Remark 8. The secret sharing protocol achieving the maximal secrecy rate $\widetilde{R}^*_{avg}(n,\varepsilon,\delta)$ presented in [HTW16] requires two-way communication over the public channel. This is in contrast to the achievability proof in the asymptotic setting (Theorem 4) for which one-way communication suffices.

This limitation of requiring two-way communication in achievability proof has practical consequences, in particular in the context of applying the result to Physical Unclonable Functions as for this purpose a protocol only requiring one-way communication is necessary.

5 PUF-Based Tamper Protection Foil

Going from the pure PUF-based key generation scheme to tamper protection brings additional requirements for the PUF. The combination of enrollment phase and reconstruction phase using HDAs as proposed in Section 3.5 for the combination of legitimate user and a physical attacker is related to secure communication over a discrete memoryless degraded wiretap channel or a discrete memoryless wiretap channel depending on the chosen attacker models that are specified in the following.

5.1 Attacker Model

Aside from the standard attack vectors of key generation schemes with analog PUFs, like helper data leakage, in addition the physical tampering aims at obtaining secret information, which needs to be prevented by the designer.

Considering all attack vectors, the remaining security of the system needs to exceed a specified minimum security level such as 120 bit, as currently recommended by the BSI [BSI24]. In this work, we chose to investigate 128, 192 and 256 Bit security levels as they are common in many cryptographic applications.

Leakage through Helper Data Since early approaches like the fuzzy commitment or fuzzy extractor [DRS04b], helper data leakage is a topic to be considered when designing secure key generation with noisy secrets, and in particular for PUFs [DGSV15] when imperfections come into play [DGV+16]. For the remainder of the this work, we assume a random number with full entropy for selecting the codeword and refer to the mentioned related works to design helper data schemes that avoid leakage through $\widetilde{W}^n$ by the helper data algorithm. In addition to $\widetilde{W}^n$, the quantization helper data W^n needs to be considered for analog PUFs. We therefore apply the zero leakage helper data generation proposed in Section 3.6 for W^n throughout this work.

Physical Tampering As physical tamper protection aims to withstand attackers being able to use sophisticated tools, a wide range of attacks needs to be considered [Wei00, Imm19, GSHO21, SI23], where a special emphasis is put to physical drilling, that affects only very small areas.

As reference, for the system proposed in [IOK+18], drilling with a conventional drill bit with diameter $> 300\mu m$ fully destroys one electrode in both layers, such that 23 out of 128, or 18% of the capacitance values are destroyed. Attacks during operation can be detected with the integrated run-time tamper detection measures to bring the system in a secure state. In contrast, attacks on the powered-off device are more challenging from a theoretical point of view. For identical layouts, we need to assume that the attacker knows the position of the destroyed PUF cells within the PUF response, which we model as erasures in the proposed coding schemes.

A neuralgic point for an attack lies between the measurement circuit and the key generation, as the attacker can obtain the digitized low-noise values from tapping a single wire, before the values are interpreted in the embedded key management system and the attack detection triggers. The attacker can only perform a single measurement of the PUF in this case because after this measurement the system will notice that the foil has been attacked and goes into a secure state, eliminating the possibility for further measurements. In the following, this attacker is referred to as the “digital attacker“ for brevity.

In another scenario, the attacker performs more advanced analog measurements, i.e. we assume he affords better measurement equipment. We call this kind of attacker the "analog attacker" for short. We make the very conservative assumption the attacker can measure with infinite precision, i.e. his equipment has infinitely fine quantization steps and furthermore he can perform an unlimited number of measurements. Due to the unlimited amount of measurements with uncorrelated measurement noise, the attacker is able to apply post-processing to effectively eliminate the measurement noise. However, in this scenario the attacker needs to drill multiple holes to tap multiple analog wires, potentially with a larger wire diameter instead of only one small hole to probe a digital signal. We will show in Section 6 that this attack is indeed problematic, especially for larger field sizes and hence propose to apply countermeasures on the hardware level eliminating the possibility for an attacker to perform those advanced measurements. A profound reasoning for the amount of PUF cells being destroyed by this attack is out of scope of this work. It is reasonable though to assume that a significant extra proportion of the foil needs to be destroyed compared to the digital attacker to mount such an attack. In our examples, we used twice the amount of destroyed capacitances compared to the digital attacker.

Within the helper data algorithm the ECC enables recovering the key from an erroneous PUF measurement as long as the number of errors is sufficiently small. Designers therefore need to take care that an attacker cannot retrieve information about the secret key using the redundancy inflicted by the ECC even though the attacker has to destroy a fraction of the PUF cells to measure the PUF. Investigating fundamental limits of this problem is the major contribution of this work.

In addition to the possibility of wiretapping internal signals, the cable connections through the foil that could reveal timing or global power side channels need to be taken into account. This topic will not be considered in the scope of this work and needs to be addressed individually when designing the host system.

5.2 Enrollment and Reconstruction Phase for the Foil PUF, Legitimate User

The PUF considered in this work is established by a foil consisting of electrodes forming a mesh of capacitances. The purpose of this PUF is twofold. First it is used for storing a cryptographic key in a secure manner and secondly it shall protect the components within its inside. An attacker trying to perform a side channel attack is required to measure parameters related to the implementation and the foil prevents him from accessing critical hardware. As composition of physical variations, the capacitances can be modelled as Gaussian random variables [IOK+18]. While the work in [GXKF22] focuses on the reliability side with a strong focus on the specific implication and its measurements, this work aims at quantifying the information theoretic security limits of this type of PUF.

The distribution of the differentially evaluated PUF response for a single cell follows a Gaussian of zero mean and variance σ_P^2, i.e. $X \sim \mathcal{N}(0, \sigma_P^2)$. We assume that the PUF values for the individual cells to be independently identically distributed (iid). We denote the output one PUF cell during the enrollment phase by the random variable X. To obtain the secret S for this cell, the measured data is quantized (see Section 3.5). The generation of the helper data $W$of a PUF is peculiar because not only shall it be of value within the reconstruction process but have the zero leakage property (Definition 4). Zero-leakage quantization helper data generation has already been discussed in Section 3.6 and we follow this approach throughout this work. In Fig. 6 the pdf of a single PUF cell is depicted. The red dot represents the realization of X. In this example we used 2 bit quantization to obtain the value of the secret value of the PUF, denoted by S. The dashed lines represent the borders of the quantizer and the numbers inside the intervals represent the respective values of S, i.e. in our example $S = 3$. The location of X within the respective interval is

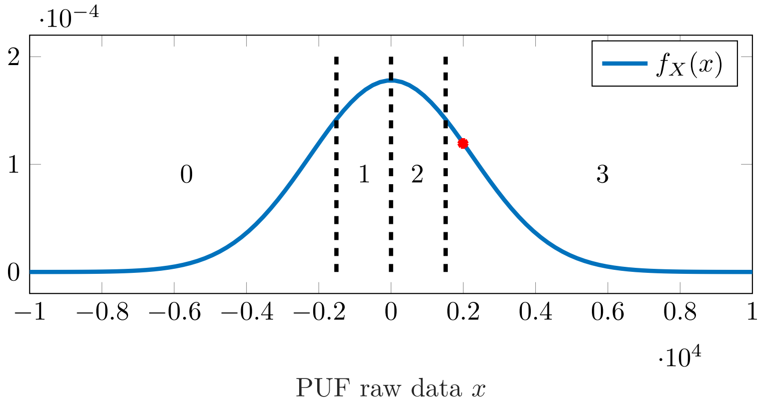

Figure 6: Enrollment phase

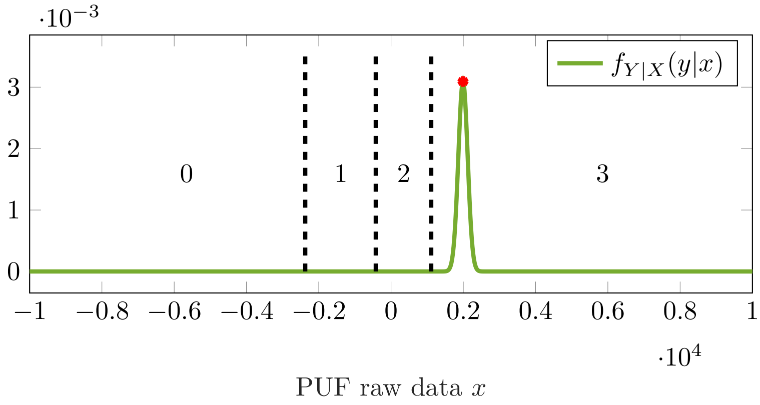

Figure 7: Reconstruction phase

then used to derive the helper data W which is later on used to find an estimate for S during the reconstruction phase.

The PUF response during the reconstruction phase is modelled as the PUF response during the enrolment phase X perturbed by additive Gaussian noise with variance σ_N^2. We denote this output by the random variable $Y = X + N$ with $N \sim \mathcal{N}(0, \sigma_N^2)$. During the reconstruction phase, Y is combined with the helper data W to output an estimate for S, in the following denoted as $\widetilde{S}$. Throughout this work we use $\sigma_P = 2241$ and $\sigma_N = 129$ which is consistent with the results in [GXKF22]. The output quantizer Q_{out} depends on the quantization helper data W and has been specified in Section 3.6. We denote the output of this quantizer by $\widetilde{S}$.

The conditional distribution of the PUF response during the reconstruction phase for the example in Fig. 6 is shown in Fig. 7. Notice that the quantization intervals during the reconstruction phase have been shifted compared to the enrollment phase. This is due to the utilization of the helper data W. By utilizing the helper data the distance between the value during the enrollment phase (the red dot) and the relevant quantization boundary for the error has increased, leading to a lower reconstruction failure probability.

Our next goal is to secure the HDA for the foil PUF against the attack scenarios mentioned in Section 5.1. Section 3.7 discusses the connection between secret sharing with common randomness and HDAs for PUFs. However, it has also been pointed out that

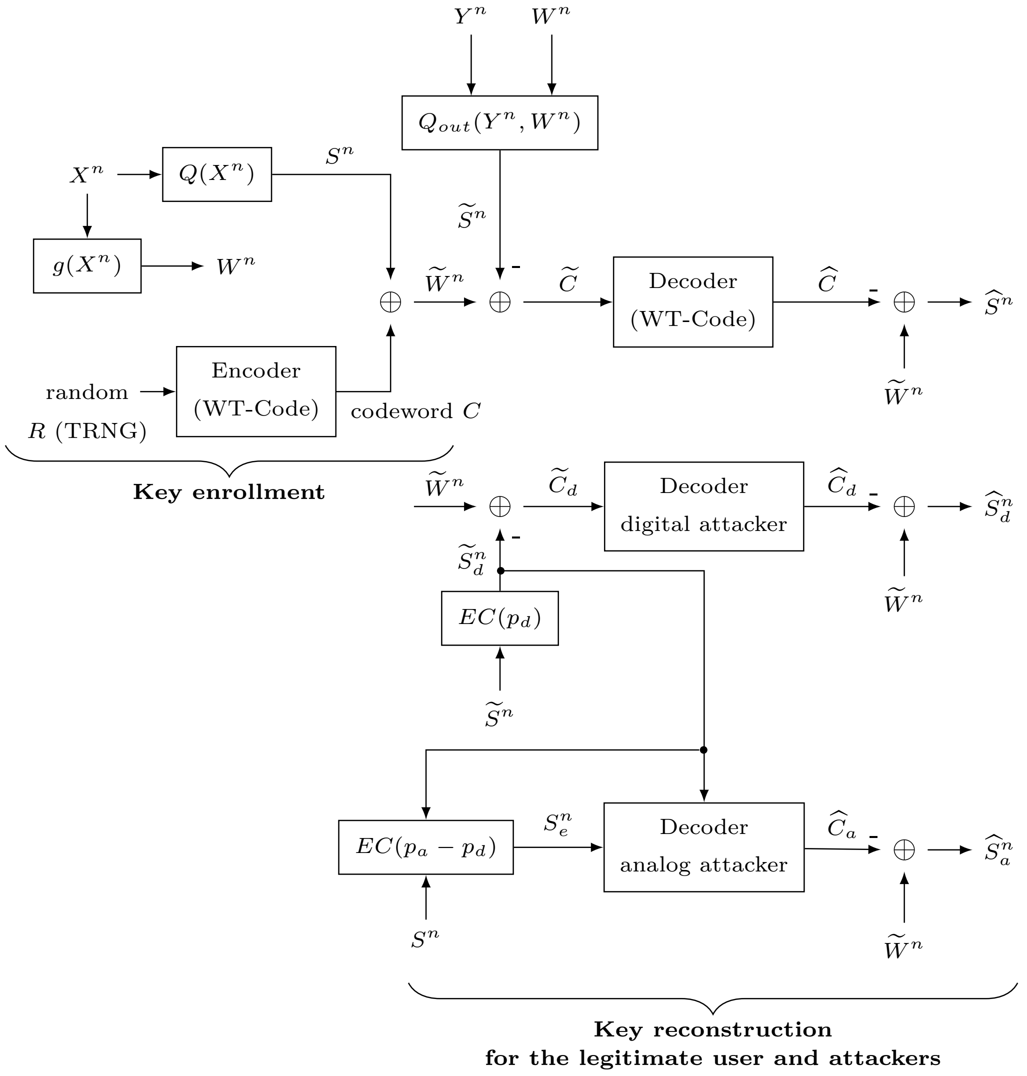

Figure 8: Key enrollment and key reconstruction for legitimate user, digital and analog attacker

the secret sharing protocol used to establish Theorem 8 requires two-way communication, thereby making it unapplicable in the PUF setting. Hence, in the following we investigate secret sharing using common randomness restricted to one-way communication in the finite blocklength regime.

6 Secret Key Capacities for the Foil PUF

In this section, we establish the connection between enrollment and reconstruction phase for the legitimate user combined with the attacker models introduced in Subsection 5.1 and the secret key generation using correlated sources. The enrollment and reconstruction phase during normal operating conditions as well as for the two mentioned attack scenarios are depicted in Fig. 8.

We first tackle the problem of determining the maximal code rate of the ECC for a HDA providing security against the attacks described in Section 5.1. Essentially, this code rate is related to the hardware complexity and the security level of the PUF as outlined in Section 3.4.

Updating Fig. 5 to also include the attacker models discussed in Section 5.1 lead to the block diagram shown in Fig. 8. Depending on whether we exclude the analog attacker by preventing the necessary measurements on a hardware level as proposed in Section 5.1 only the variables connected to the decoder for the digital attacker are of concern or also the parts associated with the analog attacker. To keep the analysis of the model simple, we take the approach of examining the scenario where only the digital attacker needs to be considered first. The changes necessary to also integrate the analog measurements of the PUF then build from there.

We observe that for the reconstruction phase the legitimate user measures Y^n and utilizing the quantization helper data W^n obtains an estimate for S^n referred to as $\widetilde{S}^n$. In contrast via the probing the digital measurement line the attacker obtains the same measurement with additional erasures $\widetilde{S}^n_d$. By a similar argument to the one in Section 3.7 we have that the resulting helper data scheme's task is to implement a secret sharing algorithm using a source of common randomness. The source of common randomness is formed by the joint probability distribution $P_{S,\widetilde{S},\widetilde{S}_d}$. In the secret sharing problem, the first terminal Alice gets access to S^n and wants to agree on a key with Bob who gets $\widetilde{S}^n$, while the eavesdropper Eve obtains $\widetilde{S}^n_d$. As described in Section 3.3, Alice and Bob next exchange messages over a public channel and aim to agree on a secret key in a secure manner, i.e. in a way such that it is impossible for Eve to obtain information about the key. Interpreting the HDA as a way to integrate a secret sharing protocol, the information shared over the public channel are the quantization helper data W^n and the reconstruction helper data $\widetilde{W}^n$. Alice's and Bob's goal is to agree on a key which is as large as possible, thereby maximizing the entropy of the key for some arbitrary but fixed n. Results on the secret key capacity (asymptotic setting as $n \to \infty$) have been recapitulated in Section 3.3 (Theorem 4) and applying these results to the HDA scheme in Fig. 8 leads to the result in Theorem 9.

Definition 6. We consider an HDA with q input quantization levels and an attacker trying to obtain the secret established during the enrollment process. The key capacity $C^q_{key,attacker}$ in this setup is the maximal asymptotic code rate for which a wiretap code exists such that the block error probability for the legitimate user is arbitrarily small and the system achieves an arbitrarily high secrecy level as the blocklength n goes to infinity.

Theorem 9. *The key capacity $C^q_{key,dig}$ for the foil PUF and the digital attacker is optimal in the sense that it enables a code rate for the ECC that is equal to the secret key capacity of the secret sharing problem with common randomness specified by the source $P_{S,\widetilde{S},\widetilde{S}_d}$. More specifically, the code rate $C^q_{key,dig}$ is defined by*

$$C^q_{key,dig} := \max_{input\ quantizer} I(S;\widetilde{S}|W)p_d \ . \tag{30}$$

Proof. First of all it is obvious that it is impossible to achieve a rate higher than the secret key capacity of the secret sharing problem using common randomness specified by the joint distribution $P_{S,\widetilde{S},\widetilde{S}_d}$.

For the achievability consider the codewords of the ECC to be sampled randomly and iid from the uniform distribution over the range of S, denoted by $\mathcal{S}$. The HDA in Fig. 8 precisely implements the construction outlined in the achievability proof sketch of Theorem 4. Hence, due to the fact that $\widetilde{S}_e$ is defined to be the output of an erasure channel having erasure probability p_d.

By the proof sketch for the achievability of Theorem 4 and by identifying $S \equiv X$,

$Y \equiv \widetilde{S}$ and $Z \equiv \widetilde{S}_d$, we have according to Theorem 4

$$\begin{aligned}\widetilde{C}_S &= I(X;Y) - I(X;Z) = I(S;\widetilde{S}|W) - I(S;\widetilde{S}_e|W) \\ &= I(S;\widetilde{S}|W) - I(S;\widetilde{S}|W)(1-p_d) = I(S;\widetilde{S}|W)p_d \ . \end{aligned} \tag{31}$$

Furthermore, compared to the secret sharing using common randomness problem introduced in Section 3.3 we have the additional freedom of choosing the input quantizer. This extra degree of freedom can be exploited since Equation (31) is valid for any input quantizer Equation (30) follows, completing the proof. ■

In order to compute this capacity the first step is to analyze the relation between S and $\widetilde{S}$ conditioned on the knowledge of the helper data W. The distribution of S follows from the distribution of the PUF during the enrollment phase, i.e. from the distribution of X, and from the choice of the input quantizer. It is complicated to perform the maximization in Theorem 9 because the channel matrix is changing according to the input quantizer as its choice influences the output quantizer (see Remark 7). The following observation is helpful though.

Lemma 2. *For the channel resulting in the concatenation of enrolment and reconstruction phase at the legitimate user it holds that*

$$I(S;\widetilde{S}|W) = I(X;\widetilde{S}|W) \tag{32}$$

Proof. By using the chain rule of mutual information it holds that

$$\begin{aligned} I(X,S;\widetilde{S}|W) &= I(S;\widetilde{S}|W) + I(X;\widetilde{S}|W,S) &(33)\\ &= I(X;\widetilde{S}|W) + I(S;\widetilde{S}|W,X) \ . &(34)\end{aligned}$$

It holds that $I(X;\widetilde{S}|W,S) = 0$ and $I(S;\widetilde{S}|W,X) = 0$ because (W,S) uniquely determines X and X determines S and the statement of the Lemma follows. ■

Since Lemma 2 shows that we can focus on $I(X;\widetilde{S}|W)$ and furthermore

$$\begin{aligned} I(X;\widetilde{S}|W) &= H(\widetilde{S}|W) - H(\widetilde{S}|W,X) &(35)\\ &= H(\widetilde{S}|W) - H(\widetilde{S}|X) \ , &(36)\end{aligned}$$

where the last equality follows because W is a function of X. We observe that the quantity that we are interested in is the difference between the uncertainty of $\widetilde{S}$ given the publicly available helper data W compared to the uncertainty of $\widetilde{S}$ given the private PUF response during the enrollment phase X.

Next we deal with the additional analog attacker. This attacker not only gets access to the digitized measurement of the internal circuitry of the PUF after the erasure channel $EC(p_d)$ but also to an analog measurement of the foil PUF passed through another erasure channel. This channel inflicts an additional fraction of $(p_a - p_d)$ errors into the positions that have not yet been erased by the hole necessary for the digital measurement. Hence, this erasure channel needs to access which cells have been erased by the digital attacker and this is reflected in the additional input $\widetilde{S}_d^n$. Its only purpose is to declare which cells have already been erased in this context.

The attacker's decoder however can utilize the information that $\widetilde{S}_d^n$ provides. This information is only useful for reconstructing the values that have been erased by the additional erasures because the number of measurements for the analog attacker is unlimited and hence unerased symbols are perfect reconstructions of the enrollment value S^n.

Theorem 10. *For the key capacity $C^q_{key,ana}$ of the foil PUF and the analog attacker it holds that*

$$\max_{input\ quantizer} I(S;\widetilde{S}|W)(1-p_a+p_d)-H(S)(1-p_a) \leq C^q_{key,ana} \leq \max_{input\ quantizer} I(S;\widetilde{S}|W)p_d \ . \tag{37}$$

Proof. We first prove the lower bound. Using Theorem 4 we have that

$$C^q_{key} \geq I(S;\widetilde{S}|W) - I(S;\widetilde{S}_d,\widetilde{S}_a|W) \ . \tag{38}$$

By the definition of mutual information it holds that

$$I(S;\widetilde{S}_d,\widetilde{S}_a|W) = H(S) - H(S|\widetilde{S}_d,\widetilde{S}_a,W) \tag{39}$$

because $H(S|W) = H(S)$ due to the zero leakage condition and furthermore we observe that

$$\begin{aligned} H(S|\widetilde{S}_d,\widetilde{S}_a,W) &= H(S|\widetilde{S}_d = 0,\widetilde{S}_a = E,W)P_{\widetilde{S}_d}(0)(p_a-p_d) \\ &+ \dots \\ &+ H(S|\widetilde{S}_d = |\mathcal{S}|,\widetilde{S}_a = E,W)P_{\widetilde{S}_d}(|\mathcal{S}|)(p_a-p_d) \\ &+ H(S|\widetilde{S}_d = E,\widetilde{S}_a = E,W)\,p_d \\ &= H(S|\widetilde{S},W)(p_a-p_d) + H(S)p_d \ . \end{aligned} \tag{40}$$

Plugging these simplifications into the right hand side of (38), we have

$$H(S)p_d - H(S|\widetilde{S},W)(1-p_a+p_d) \tag{41}$$

which is equivalent to the lower bound in the theorem after some simple algebraic steps and maximizing over the choice of the input quantizer.

The upper bound follows trivially because the analog attacker is stronger than the digital attacker. However, we also show that the upper bound cannot be trivially improved by the well known upper bound

$$I(S;\widetilde{S}|\widetilde{S}_d,\widetilde{S}_a,W) = H(S|\widetilde{S}_d,\widetilde{S}_a,W) - H(S|\widetilde{S},\widetilde{S}_d,\widetilde{S}_a,W) \ . \tag{42}$$

The first entropy in Equation (42) has already been simplified within this proof. Hence, we only need to simplify $H(S|\widetilde{S},\widetilde{S}_d,\widetilde{S}_a,W)$. Observe that

$$H(S|\widetilde{S},\widetilde{S}_d,\widetilde{S}_a,W) = H(S|\widetilde{S},\widetilde{S}_a = E,W)p_a = H(S|\widetilde{S})p_a \tag{43}$$

and therefore

$$I(S;\widetilde{S}|\widetilde{S}_d,\widetilde{S}_a,W) = I(S;\widetilde{S}|W)p_d \ . \tag{44}$$

■

Evaluation of Theorem 9 and Theorem 10 leads to the achievability results presented in Tables 1 and 2. We assumed that the analog attacker has to destroy twice as much of the foil as the digital attacker has to. In Table 1 we took the conservative approach of assuming that the digital attacker destroys 10% of the PUF cells, i.e. $p_d = 0.1, p_a = 0.2$. Table 2 shows the results for $p_d = 0.18, p_a = 0.36$, which is the value corresponding to destroying one electrode in both layers of the foil as discussed in [IOK+18] considering a hole diameter $> 300\mu m$.

In both tables we considered various numbers of quantization levels and input quantizer strategies. From the proofs of Theorem 9 and Theorem 10 it is easy to see that choosing a suboptimal input quantizer still gives an achievable lower bound on the secret key rate.

Table 1: Achievable asymptotic rates for the PUF-channel for weak and strong attackers, different input quantization alphabets and quantization strategies, erasure probability $p_d = 0.1$ for the digital attacker and $p_a = 0.2$ for the analog attacker, $\sigma_P = 2241, \sigma_N = 129$

quantizer levels	equidist. digital	equidist. analog	equiprob. digital	equiprob. analog	optimized digital	optimized analog
2	0.1	0.1	0.1	0.1	0.1	0.1
4	0.117	0.117	0.2	0.2	0.2	0.2
8	0.196	0.196	0.298	0.282	0.299	0.294
16	0.291	0.291	0.356	0	0.363	0.327
32	0.376	0.269	0.382	0	0.382	0.311
64	0.389	0	0.398	0	0.398	0
128	0.404	0	0.406	0	0.406	0
256	0.410	0	0.409	0	0.410	0

Table 2: Achievable asymptotic rates for the PUF-channel for weak and strong attackers, different input quantization alphabets and quantization strategies, erasure probability $p_d = 0.18$ for the digital attacker and $p_a = 0.36$ for the analog attacker, $\sigma_P = 2241, \sigma_N = 129$

quantizer levels	equidist. digital	equidist. analog	equiprob. digital	equiprob. analog	optimized digital	optimized analog
2	0.18	0.18	0.18	0.18	0.18	0.18
4	0.211	0.211	0.36	0.36	0.36	0.36
8	0.353	0.353	0.536	0.524	0.537	0.533
16	0.523	0.523	0.640	0.356	0.654	0.600
32	0.677	0.591	0.687	0	0.688	0.613
64	0.700	0.061	0.716	0	0.716	0.061
128	0.727	0	0.731	0	0.731	0
256	0.738	0	0.737	0	0.738	0

In this work, we consider equidistant and equiprobable input quantization. Furthermore, we use an optimization algorithm aiming to find the input quantizer maximizing the achievable secret key rate according to Theorem 9 and Theorem 10.

The results show that in case we have to cope with the digital attacker that the equiprobable input quantization performs better than equidistant quantization. For the analog attacker the opposite is the case as we increase the number of quantization levels. For a small number of quantization levels equiprobable quantization still performs better.

The reason for that behaviour is that the analog attacker obtains a perfect duplicate of the PUF cell measured during enrollment in case the respective cell is not erased by the holes he is required to drill to perform measurements. Especially if the measurement during the reconstruction by the legitimate user is unreliable, the analog measurements provide valuable additional information to the analog attacker that the legitimate user does not have. The input quantization has a substantial effect on the reliability of the measurement during reconstruction. Once an interval length at the output quantizer (influenced by the input quantizer) is below a certain threshold a measurement result of this value becomes unreliable. In this case the additional information provided by the analog measurement substantially increases due to the insecurity of the reconstruction value for the legitimate user.

This explains why equidistant input quantization performs better for a larger number of quantization levels if the analog attacker is considered. For a small amount of quantization levels the intervals are anyway large enough and the dominant factor is the entropy of the input, which is obviously maximized for the uniform input distribution achieved by the equiprobable input quantizer. For the digital attacker we do not have this trade-off and hence equiprobable quantization always performs better than equidistant quantization.

7 Secret Sharing using One-Way Communication for Finite Lengths

For this section we stick to the notation used in Section 3.3 that is commonly used in secret sharing with common randomness. In particular X, Y and Z are not related to the foil PUF in particular but rather form a general source of randomness via the pmf P_{XYZ}. After this section X shall again be considered the analog measurement during enrollment and Y shall again be considered the analog measurement during reconstruction.

Because of this inherent limitation of the construction used in [HTW16], we take the approach of utilizing the construction in the achievability proof sketch of Theorem 4 also in the finite blocklength case.

Theorem 11. *For the secret key rate with maximum secrecy δ and average error probability ε it holds that*

$$\widetilde{R}^*_{max}(n,\varepsilon,\delta) \geq \widetilde{C}_S - \sqrt{\frac{V_1}{n}}Q^{-1}(\varepsilon) - \sqrt{\frac{V_2}{n}}Q^{-1}(\delta) + \mathcal{O}\left(\frac{\log(n)}{n}\right) \quad , \tag{45}$$

where

$$V_1 = \sum_{\substack{x\in\mathcal{X}\\ y\in\mathcal{Y}}} P_{XY}(x,y)\log_2^2\left(\frac{P_{XY}(x,y)}{\frac{1}{|\mathcal{X}|}P_Y(y)}\right) - D\left(P_{XY}||P_{\mathcal{X}}^{unif}P_Y\right)^2 \tag{46}$$

and

$$V_2 = \sum_{\substack{x\in\mathcal{X}\\ z\in\mathcal{Z}}} P_{XZ}(x,z)\log_2^2\left(\frac{P_{XZ}(x,z)}{\frac{1}{|\mathcal{X}|}P_Z(z)}\right) - D\left(P_{XZ}||P_{\mathcal{X}}^{unif}P_Z\right)^2 \tag{47}$$

with $P_{\mathcal{X}}^{unif}$ denoting the uniform distribution over the input alphabet $\mathcal{X}$. This rate can be achieved using one-way communication over the public channel.

Proof. As in the achievability proof sketch for Theorem 4, we use a secrecy codebook for the degraded wiretap channel sampled according to a distribution P_U, where in this particular case we define P_U to be uniform over the alphabet $\mathcal{X}$. We mask the codeword symbolwise by computing $u_i + x_i$ using the common randomness provided by the source P_{XYZ} and send the result over the public channel to Bob. Again the artificially created wiretap channel with input U and outputs $(U + X, Y)$ and $(U + X, Z)$ can be used to show achievability results for secret sharing using common randomness, this time in the finite blocklength regime. To compute the achievable secret key rates we apply Theorem 7. Notice that the secrecy definitions for (degraded) wiretap channels and secret sharing using common randomness match each other. Hence, the secrecy condition of the secrecy code implies the secrecy definition for secret sharing using common randomness.

For the first channel dispersion term V_1 the legitimate user channel from U to $(U+X, Y)$ is relevant. Hence, we have

$$V_1 = \sum_{u\in\mathcal{X}} P_U(u)\left(\sum_{\substack{\tilde{x}\in\mathcal{X}\\ y\in\mathcal{Y}}} P_{U\oplus X,Y|U}(\tilde{x},y|u)\log_2^2\left(\frac{P_{U\oplus X,Y|U}(\tilde{x},y|u)}{P_{U\oplus X,Y}(\tilde{x},y)}\right)\right.$$
$$\left. - D\left(P_{U\oplus X,Y|U=u}||P_{U\oplus X,Y}\right)^2\right) \ . \tag{48}$$

Next we analyze the terms $P_{U\oplus X,Y|U}(\tilde{x},y|u)$ and $P_{U\oplus X,Y}(\tilde{x},y)$. We have

$$P_{U\oplus X,Y|U}(\tilde{x},y|u) = P_{U\oplus X|U}(\tilde{x}|u)P_{Y|U\oplus X,U}(y|\tilde{x},u) = P_X(\tilde{x}-u)P_{Y|X}(y|\tilde{x}-u) \tag{49}$$

because U is uniformly distributed over $\mathcal{X}$ and independent of X, Y.

Furthermore, it holds that

$$P_{U\oplus X,Y}(\tilde{x},y) = P_{U\oplus X}(\tilde{x})P_{Y|U\oplus X}(y|\tilde{x}) = \frac{1}{|\mathcal{X}|}P_Y(y) \tag{50}$$

again because U is uniformly distributed over $\mathcal{X}$ and independent of X, Y. Notice that $U \oplus X$ is independent of Y even though X and Y may not be independent. This is basically because X is encrypted by a one-time pad using U as its key.

Equation (46) follows because the sum in the bracket goes over the entire alphabet sets $\mathcal{X}$ and $\mathcal{Y}$. Notice that the sums do not depend on the choice of U in the outer sum. The same holds for the sums defining the divergence term.

An analogous argument holds for V_2 in Equation (47). ■

Theorem 12. *Let us assume that we use the same communication protocol as in the proof of Theorem 11. In this case for the secret key rate with average secrecy δ and average error proabability ε it holds that*

$$\widetilde{R}^*_{avg} \leq \widetilde{C}_S - \sqrt{\frac{V_c}{n}}Q^{-1}(\varepsilon+\delta) + \mathcal{O}\left(\frac{\log(n)}{n}\right) \ , \tag{51}$$

where

$$V_c = \sum_{\substack{x\in\mathcal{X}\\ y\in\mathcal{Y}\\ z\in\mathcal{Z}}} P_{XYZ}(x,y,z)\log_2^2\left(\frac{P_{XYZ}(x,y,z)}{P_{XZ}(x,z)P_{Y|Z}(y|z)}\right) - D(P_{XYZ}||P_{Y|Z}P_{XZ})^2 \ . \tag{52}$$

This upper bound holds in particular for the secret sharing problem using only one-way communication over the public channel.

Proof. As mentioned in the theorem, our goal is again to use a wiretap code for the channel with uniformly sampled input U and with outputs $(U+X, Y)$ and $(U+X, Z)$. Hence, an upper bound on the secrecy rate for a wiretap code of this channel is also an upper bound on the achievable secret key rate of the communication protocol for the secret sharing problem.

Using the upper bound for wiretap codes in Theorem 7 it holds that

$$V_c = \sum_{u\in\mathcal{X}} P_U(u)\left(\sum_{\substack{\tilde{x}\in\mathcal{X}\\ y\in\mathcal{Y}\\ z\in\mathcal{Z}}} P_{U\oplus X,Y,Z|U}(\tilde{x},y,z|u)\log_2^2\left(\frac{P_{U\oplus X,Y,Z|U}(\tilde{x},y,z|u)}{P_{U\oplus X,Z|U}(\tilde{x},z|u)P_{U\oplus X,Y|U\oplus X,Z}(\tilde{x},y|\tilde{x},z)}\right)\right.$$
$$\left. - D(P_{U\oplus X,Y,Z|U=u}||P_{U\oplus X,Y|U\oplus X,Z}P_{U\oplus X,Z|U=u})^2\right) \ . \tag{53}$$

To simplify Equation (53) we observe

$$\begin{aligned} P_{U\oplus X,Y,Z|U}(\tilde{x},y,z|u) &= P_{U\oplus X|U}(\tilde{x}|u)P_{Y|U\oplus X,U}(y|\tilde{x},u)P_{Z|Y,U\oplus X,U}(z|y,\tilde{x},u) \\ &= P_X(\tilde{x}-u)P_{Y|X}(y|\tilde{x}-u)P_{Z|Y,X}(z|y,\tilde{x}-u) \ , \end{aligned} \tag{54}$$

$$P_{U\oplus X,Z|U}(\tilde{x},z|u) = P_{U\oplus X|U}(\tilde{x}|u)P_{Z|U\oplus X,U}(z|\tilde{x},u) = P_X(\tilde{x}-u)P_{Z|X}(z|\tilde{x}-u) \tag{55}$$

and

$$P_{U\oplus X,Y|U\oplus X,Z}(\tilde{x},y|\tilde{x},z) = P_{Y|Z}(y|z) \ . \tag{56}$$

The basic idea to show all of these simplifications is to use the facts that U is uniformly distributed over $\mathcal{X}$ and independent of X, Y, Z and that $U\oplus X$ is independent of Y and Z because X is encrypted by a one-time pad using the uniformly distributed U as a key.

As in the proof of Theorem 11 the inner sum in Equation (53) goes over the entire alphabet sets $\mathcal{X}, \mathcal{Y}$ and $\mathcal{Z}$. Hence, the sums do not depend on the specific value of U since it only leads to an index shift within the sum. The same argument holds for the divergence term. ■

Corollary 2. *Let us consider a source of common randomness following a joint pmf P_{XYZ} for which it holds that $X -\!\circ\!- Y -\!\circ\!- Z$. Then it holds that the upper bound on the maximal secrecy rate for wiretap codes in Theorem 7 is not tight for the degraded wiretap channel formed by the uniformly sampled input U (over $\mathcal{X}$), legitimate user output $(U\oplus X, Y)$ and eavesdropper output $(U\oplus X, Z)$.*

Proof. As already emphasized in Remark 8 the communication protocol over the public channel achieving the secret key rate in Theorem 8 requires two-way communication over the public channel while the construction of the upper bound in Theorem 12 only requires one-way communication. Hence, in order to prove the corollary, it suffices to show that the upper bound in Theorem 12 exceeds the maximal secret key rate in Theorem 8. We will show in the following that this indeed holds.

The difference between the rates in Theorems 12 and 8 lies in the channel dispersion terms V_c and V_c', respectively.

Notice that the sums involving the $\log_2^2$ terms are equivalent. Hence, we focus on the divergence terms in both equations. We have

$$\begin{aligned}
D(P_{XYZ}||P_{Y|Z}P_{XZ})^2 &= \left(\sum_{\substack{x\in\mathcal{X}\\ y\in\mathcal{Y}\\ z\in\mathcal{Z}}} P_{XYZ}(x,y,z)\log_2\left(\frac{P_{XYZ}(x,y,z)}{P_{Y|Z}(y|z)P_{XZ}(x,z)}\right)\right)^2 \\
&= \left(\sum_{x\in\mathcal{X}} P_X(x) \sum_{\substack{y\in\mathcal{Y}\\ z\in\mathcal{Z}}} P_{YZ|X}(y,z|x)\log_2\left(\frac{P_{YZ|X}(y,z|x)}{P_{Y|Z}(y|z)P_{Z|X}(z|x)}\right)\right)^2
\end{aligned} \tag{57}$$

and

$$\begin{aligned}
&\sum_{x\in\mathcal{X}} P_X(x)D(P_{YZ|X=x}||P_{Y|Z}P_{Z|X=x})^2 \\
&= \sum_{x\in\mathcal{X}} P_X(x)\left(\sum_{\substack{y\in\mathcal{Y}\\ z\in\mathcal{Z}}} P_{YZ|X}(y,z|x)\log_2(\frac{P_{YZ|X}(y,z|x)}{P_{Y|Z}(y|z)P_{Z|X}(z|x)})\right)^2 .
\end{aligned} \tag{58}$$

Because $f(x) = x^2$ is a strictly convex function we have that

$$D(P_{XYZ}||P_{Y|Z}P_{XZ})^2 > \sum_{x\in\mathcal{X}} P_X(x)D(P_{YZ|X=x}||P_{Y|Z}P_{Z|X=x})^2 \tag{59}$$

and hence the statement follows. ■

Remark 9. Corollary 2 shows for sources of common randomness P_{XYZ} for which $X \multimap\!\!- Y \multimap\!\!- Z$ that the upper bound obtained in Theorem 12 is strictly larger than the rate obtained in Theorem 8. The secret key rate of Theorem 8 also provides an upper bound on the secret key rate for protocols with one-way communication. Notice though that Theorem 8 requires the random variables X, Y, Z to be connected via a Markov chain while for Theorem 12 can also be applied if this is not the case.

Equipped with the knowledge of this section we next tackle the problem of securing the HDA against the attack scenarios mentioned in Section 5.1 for finite lengths.

8 Achievability and Converse Bounds for Finite Lengths

The results presented in the previous section are for the asymptotic setting in the sense that they hold if the number of capacitive cells in the PUF goes to infinity and the fraction of the erasures for the attacker are determined by the erasure channel parameters p_d and p_a for the digital and analog attacker model. Of course, as a first order approximation one could compute the key capacity C^q_{key} and estimate the dimension of the secrecy code for the resulting (degraded) wiretap channel by $C^q_{key}n$. As for the channel models discussed in Section 4 to estimate the code dimension in that way is highly inaccurate if the blocklength is of small to moderate size. In general the achievable code dimension is of significantly smaller size. We note that the erasure channel assumption is only an approximation of the reality since for our attacker model the number of erasures is assumed to be constant and equal to p_dn or p_an for the digital and the analog attacker model, respectively. Nevertheless, we consider the results obtaining from this to be rather accurate because these values are equal to the expectation for the erasure channels. Notice here that erasures in the digital attacker model imply erasures in the analog attacker model. The erasures for the analog attacker are then added on top.

Using Theorem 11 to estimate the achievable code rate leads to a much more precise estimate for small to moderate blocklengths. Similarly, applying Theorem 12 is much more precise to estimate a converse result on the secret key rate for the HDA. However, using Theorem 8 enables us to find an even tighter converse in case we only need to protect the design against the digital attacker.

The following observation is helpful and used implicitly throughout this section. Empirically, we found that $I(S;\widetilde{S}|W) \approx I(S;\widetilde{S})$. Basically we observed no difference between these quantities even though we were not able to formally prove this statement. Notice that this does not mean that the helper data has not been utilized on the right hand side of the equation. Rather it means that the channel matrix on the right hand side is constructed by averaging over all possible values of W rather than constructing the channel matrix for each realization w, computing $I(S;\widetilde{S}|W=w)$ and then averaging with respect to $f_W(w)$. This effect is desirable as it shows that it is sufficient to construct a single codebook independent of the helper data W, rather than constructing different codebooks for different values of W. This behaviour makes perfect sense considering that $\widetilde{S}$ is an approximation of S and W is independent of S due to the zero leakage condition (see Section 3.6).

The potential inaccuracy by assuming $I(S;\widetilde{S}) = I(S;\widetilde{S}|W)$ only induces a rate penalty for the scheme. In terms of security this is not a problem as W is still perfectly utilized and the codebook is chosen by the legitimate users. Furthermore, it makes the application of Theorem 7 significantly easier.

In the following, we are investigating which finite rates can be achieved by HDAs with different parameters for both attacker models. We are in particular interested in this rate as the code dimension in the HDA has to be at least as large as the targeted security level. A lower bound on the code rate (which also depends on the blocklength for finite n) leads to an upper bound on the required amount of capacitive cells while a lower bound can be used to prove impossibility results, i.e. it enables to show that a certain amount of cells is absolutely required for target values in terms of reliability and security of the PUF.

As in Section 6 we first consider the HDA scheme only covering the digital attacker.

8.1 Digital Attacker

Theorem 13. *The maximal achievable rate $\widetilde{R}^{key,dig}_{max}(n,\varepsilon,\delta)$ for the HDA achieving a maximum secrecy level δ against the digital attacker described in Section 5.1 with error probability ε during regular device operation is lower bounded by*

$$\widetilde{R}^{key,dig}_{max}(n,\varepsilon,\delta) \geq R^{q}_{asymp,dig} - \sqrt{\frac{V_1}{n}}Q^{-1}(\varepsilon) - \sqrt{\frac{V_2}{n}}Q^{-1}(\delta) + \mathcal{O}\left(\frac{\log(n)}{n}\right) \quad , \tag{60}$$

where

$$R^{q}_{asymp,dig} := I(S;\widetilde{S})p_d \quad , \tag{61}$$

$$V_1 = \sum_{\substack{s\in\mathcal{S}\\ \tilde{s}\in\mathcal{S}}} P_{S,\widetilde{S}}(s,\tilde{s}) \log_2^2\left(\frac{P_{S,\widetilde{S}}(s,\tilde{s})}{\frac{1}{|\mathcal{S}|}P_{\widetilde{S}}(\tilde{s})}\right) - D(P_{S,\widetilde{S}}||P^{unif}_{\mathcal{S}}P_{\widetilde{S}})^2 \tag{62}$$

and

$$V_2 = \sum_{\substack{s \in \mathcal{S} \\ \tilde{s} \in \mathcal{S}}} P_{S,\widetilde{S}}(s,\tilde{s})(1-p_d) \log_2^2 \left(\frac{P_{S,\widetilde{S}}(s,\tilde{s})}{\frac{1}{|\mathcal{S}|} P_{\widetilde{S}}(\tilde{s})} \right) + \sum_{s \in \mathcal{S}} P_S(s) p_d \log_2^2 \left(\frac{P_S(s)}{\frac{1}{|\mathcal{S}|}} \right)$$
$$- \left[(1-p_d) D \left(P_{S,\widetilde{S}} || P_{\mathcal{S}}^{unif} P_{\widetilde{S}} \right) + p_d D \left(P_S || P_{\mathcal{S}}^{unif} \right) \right]^2 \quad . \tag{63}$$

Proof. The statement follows from Theorem 11 by setting $S \equiv X$, $\widetilde{S} \equiv Y$ and $\widetilde{S}_d \equiv Z$. V_1 is directly obtained by plugging in the respective random variables. For the computation of V_2 we note that the probability that $P_{\widetilde{S}_d|S}(E|s) = p_d$ irrespective of s. In case no erasure occurs it holds that $P_{\widetilde{S}_d|S}(\tilde{s}|s) = P_{\widetilde{S}|S}(\tilde{s}|s)(1-p_d)$. The rest follows easily from the definition of V_2 in Theorem 11. ■

Theorem 14. *The secret key rate for a source of common randomness with distribution $P_{S,\widetilde{S},\widetilde{S}_d}$ is*

$$\widetilde{R}^*_{avg}(n,\varepsilon,\delta) = C_S - \sqrt{\frac{V_c'}{n}} Q^{-1}(\varepsilon+\delta) + \mathcal{O}\left(\frac{\log(n)}{n} \right) \quad , \tag{64}$$

where

$$V_c' = \sum_{s \in \mathcal{S}} P_S(s) \sum_{\tilde{s} \in \mathcal{S}} P_{\widetilde{S}|S}(\tilde{s}|s) p_d \log_2^2 \left(\frac{P_{\widetilde{S}|S}(\tilde{s}|s)}{P_{\widetilde{S}}(\tilde{s})} \right) - \sum_{s \in \mathcal{S}} P_S(s) p_d^2 D \left(P_{\widetilde{S}|S=s} || P_{\widetilde{S}} \right)^2 \quad . \tag{65}$$

Furthermore, it holds that

$$\widetilde{R}^*_{avg}(n,\varepsilon,\delta) \geq \widetilde{R}^{key}_{avg}(n,\varepsilon,\delta) \quad , \tag{66}$$

where $\widetilde{R}^q_{avg,dig}(n,\varepsilon,\delta)$ denotes the maximal achievable rate for the HDA achieving an average security level δ against the digital attacker

Proof. The statement follows from Theorem 8 by setting $S \equiv X$, $\widetilde{S} \equiv Y$ and $\widetilde{S}_d \equiv Z$. Notice that $S, \widetilde{S}, \widetilde{S}_d$ form the Markov chain $S -\!\!\circ\!\!- \widetilde{S} -\!\!\circ\!\!- \widetilde{S}_d$ which is necessary for Theorem 8 to be applicable.

By the definition of $\widetilde{S}_d$ we have that $\widetilde{S}_d$ is either equal to $\widetilde{S}$ or the erasure event E. We have that

$$\begin{aligned} P_{S,\widetilde{S},\widetilde{S}_d}(s,\tilde{s},\tilde{s}) &= P_S(s) P_{\widetilde{S}|S}(\tilde{s}|s)(1-p_d) \\ P_{S,\widetilde{S},\widetilde{S}_d}(s,\tilde{s},E) &= P_S(s) P_{\widetilde{S}|S}(\tilde{s}|s) p_d. \end{aligned} \tag{67}$$

Furthermore, it holds that $P_{\widetilde{S}|\widetilde{S}_d}(\tilde{s}|\tilde{s}) = 1$ and $P_{\widetilde{S}|\widetilde{S}_d}(\tilde{s}|E) = P_{\widetilde{S}}(\tilde{s})$.

Using these statements and applying them to Theorem 8 we observe that all terms for which $\widetilde{S}_d \neq E$ go to zero.

Equation (65) follows then by considering the remaining terms for which $\widetilde{S}_d = E$. ■

By Remark 9 we know that the upper bound given in Theorem 14 is tighter than applying Theorem 12.

Using Theorem 13 and Theorem 14 we obtain the achievability and converse results presented in Tables 3, 4, 5 and 6. When we speak of a λ Bit security level we mean that the security parameter δ satisfies $\delta \leq 2^{-\lambda}$.

Table 3: Achievability (ach.) and converse (conv.) results on the number of necessary capacitive PUF cells for $p_d = 0.1$, PUF reliability $\varepsilon = 10^{-6}$ and security levels 128, 192 and 256 bit, digital attacker, equiprobable input quantization, $\sigma_P = 2241, \sigma_N = 129$

quantizer	ach.128b	conv.128b	ach.192b	conv.192b	ach.256b	conv.256b
2	3645	1902	5499	2655	7354	3391
4	2664	1117	4025	1516	5386	1902
8	3178	850	4635	1128	6072	1398
16	5502	779	7768	1019	9977	1250
32	5390	744	7609	970	9773	1187
64	5509	726	7768	943	9968	1152

Table 4: Achievability (ach.) and converse (conv.) results on the number of necessary capacitive PUF cells for $p_d = 0.18$, PUF reliability $\varepsilon = 10^{-6}$ and security levels 128, 192 and 256 bit, digital attacker, equiprobable input quantization, $\sigma_P = 2241, \sigma_N = 129$

quantizer	ach.128b	conv.128b	ach.192b	conv.192b	ach.256b	conv.256b
2	1938	1038	2923	1454	3909	1860
4	1399	606	2113	825	2828	1038
8	1508	459	2216	612	2916	760
16	2194	420	3128	552	4042	680
32	2150	401	3064	525	3959	644
64	2179	390	3102	510	4004	625

Table 5: Achievability (ach.) and converse (conv.) results on the number of necessary capacitive PUF cells for $p = 0.1$, PUF reliability $\varepsilon = 10^{-9}$ and security levels 128, 192 and 256 bit, digital attacker, equiprobable input quantization, $\sigma_P = 2241, \sigma_N = 129$

quantizer	ach.128b	conv.128b	ach.192b	conv.192b	ach.256b	conv.256b
2	3645	2106	5499	2887	7354	3647
4	2665	1286	4026	1703	5388	2106
8	3345	1006	4834	1300	6299	1582
16	6050	940	8414	1194	10705	1437
32	5927	903	8243	1142	10488	1370
64	6068	884	8427	1114	10712	1335

Table 6: Achievability (ach.) and converse (conv.) results on the number of necessary capacitive PUF cells for $p = 0.18$, PUF reliability $\varepsilon = 10^{-9}$ and security levels 128, 192 and 256 bit, digital attacker, equiprobable input quantization, $\sigma_P = 2241, \sigma_N = 129$

quantizer	ach.128b	conv.128b	ach.192b	conv.192b	ach.256b	conv.256b
2	1938	1145	2923	1575	3909	1994
4	1400	693	2114	923	2828	1145
8	1571	539	2291	701	3002	856
16	2382	504	3350	643	4293	777
32	2334	483	3282	614	4205	740
64	2370	472	3327	599	4259	720

The results show that different amounts of input quantization intervals can significantly change the achievable rates for the foil PUF. Notice that for the achievability bounds, increasing the amount of quantization intervals does not necessarily decrease the required number of PUF cells. 2 Bit quantization is optimal from that perspective, e.g. for the setting examined in Table 3 the results show that 2664 PUF cells suffice for a security level of 128 Bit. The converse bounds decrease with an increasing the amount of quantization intervals though. From the tables we observe that an increased demand in PUF reliability only has a small influence on the required number of cells, whereas the erasure probability has a much larger impact. We also observe that higher security levels require more PUF cells as was to be expected.

8.2 Analog Attacker

In this section we basically perform a similar analysis for the analog attacker as we did for the digital attacker.

Theorem 15. *The maximal achievable rate $\widetilde{R}_{max}^{key,ana}(n,\varepsilon,\delta)$ for the HDA achieving a maximum secrecy level δ against the analog attacker described in Section 5.1 with error probability ε during regular device operation is lower bounded by*

$$\widetilde{R}_{max}^{key,ana}(n,\varepsilon,\delta) \geq R_{asymp,ana}^{q,lower} - \sqrt{\frac{V_1}{n}}Q^{-1}(\varepsilon) - \sqrt{\frac{V_2}{n}}Q^{-1}(\delta) + \mathcal{O}\left(\frac{\log(n)}{n}\right) \quad , \tag{68}$$

where

$$R_{asymp,ana}^{q,lower} := I(S;\widetilde{S}|W)(1-p_a+p_d) - H(S)(1-p_a) \quad , \tag{69}$$

$$V_1 = \sum_{\substack{s\in\mathcal{S}\\ \tilde{s}\in\mathcal{S}}} P_{S,\widetilde{S}}(s,\tilde{s})\log_2^2\left(\frac{P_{S,\widetilde{S}}(s,\tilde{s})}{\frac{1}{|\mathcal{S}|}P_{\widetilde{S}}(\tilde{s})}\right) - D(P_{S,\widetilde{S}}||P_{\mathcal{S}}^{unif}P_{\widetilde{S}})^2 \tag{70}$$

and

$$\begin{aligned}V_2 =& p_d\sum_{s\in\mathcal{S}} P_S(s)\log_2^2\left(\frac{P_S(s)}{\frac{1}{|\mathcal{S}|}}\right) + (p_a-p_d)\sum_{\substack{s\in\mathcal{S}\\ \tilde{s}\in\mathcal{S}}} P_{S,\widetilde{S}}(s,\tilde{s})\log_2^2\left(\frac{P_S(s)P_{\widetilde{S}|S}(\tilde{s}|s)}{\frac{1}{|\mathcal{S}|}P_{\widetilde{S}}(\tilde{s})}\right)\\ &+ (1-p_a)\log_2^2(|\mathcal{S}|) - \Big[\log_2(|\mathcal{S}|) + I(S;\widetilde{S})(p_a-p_d) - p_aH(S)\Big]^2\end{aligned} \tag{71}$$

Proof. To prove the statement we use Theorem 11 and set $S \equiv X$, $\widetilde{S} \equiv Y$ and $(\widetilde{S}_d, \widetilde{S}_a) \equiv Z$. The channel for the legitimate user does not change compared to the digital attacker. Hence, the dispersion term V_1 does not change compared to Theorem 13.

Therefore, we only need to provide a proof for V_2. For $\widetilde{S}_d \neq E$ and $\widetilde{S}_a \neq E$, we have that

$$\begin{aligned}P_{S,\widetilde{S}_d,\widetilde{S}_a}(s,\tilde{s},s) &= P_{S,\widetilde{S}}(s,\tilde{s})(1-p_a)\\ P_{\widetilde{S}_d,\widetilde{S}_a}(\tilde{s},s) &= P_S(s)P_{\widetilde{S}|S}(\tilde{s}|s)(1-p_a) \quad .\end{aligned} \tag{72}$$

Furthermore, for $\widetilde{S}_d = E$ and $\widetilde{S}_a = E$

$$\begin{aligned}P_{S,\widetilde{S}_d,\widetilde{S}_a}(s,E,E) &= P_S(s)p_d\\ P_{\widetilde{S}_d,\widetilde{S}_a}(E,E) &= p_d\end{aligned} \tag{73}$$

and for $\widetilde{S}_d \neq E$ and $\widetilde{S}_a = E$

$$\begin{aligned}P_{S,\widetilde{S}_d,\widetilde{S}_a}(s,\tilde{s},E) &= P_S(s)P_{\widetilde{S}|S}(\tilde{s}|s)(p_a-p_d)\\ P_{\widetilde{S}_d,\widetilde{S}_a}(\tilde{s},E) &= P_{\widetilde{S}}(\tilde{s})(p_a-p_d)\end{aligned} \tag{74}$$

Using these identities and Theorem 11 we obtain

$$\begin{aligned}V_2 =& p_d \sum_{s\in\mathcal{S}} P_S(s)\log_2^2\left(\frac{P_S(s)}{\frac{1}{|\mathcal{S}|}}\right) + (p_a-p_d)\sum_{\substack{s\in\mathcal{S}\\ \tilde{s}\in\mathcal{S}}} P_{S,\widetilde{S}}(s,\tilde{s})\log_2^2\left(\frac{P_S(s)P_{\widetilde{S}|S}(\tilde{s}|s)}{\frac{1}{|\mathcal{S}|}P_{\widetilde{S}}(\tilde{s})}\right)\\ &+(1-p_a)\log_2^2(|\mathcal{S}|) - \left[\log_2(|\mathcal{S}|) + I(S;\widetilde{S})(p_a-p_d) - p_aH(S)\right]^2 \qquad (75)\end{aligned}$$

after some elementary algebraic steps. ■

Theorem 16. *The maximal achievable rate $\widetilde{R}^{key,ana}_{avg}(n,\varepsilon,\delta)$ for the HDA achieving a average secrecy level δ against the analog attacker described in Section 5.1 with error probability ε during regular device operation is upper bounded by*

$$\widetilde{R}^{key,ana}_{avg}(n,\varepsilon,\delta) \le R^{q,upper}_{asymp,ana} - \sqrt{\frac{V_c}{n}}Q^{-1}(\varepsilon+\delta) + \mathcal{O}\left(\frac{\log(n)}{n}\right) \qquad (76)$$

with

$$R^{q,upper}_{asymp,ana} := I(S;\widetilde{S}|W)p_d \qquad (77)$$

and

$$V_c = p_d\sum_{\substack{s\in\mathcal{S}\\ \tilde{s}\in\mathcal{S}}} P_{S,\widetilde{S}}(s,\tilde{s})\log_2^2\left(\frac{P_{\widetilde{S}|S}(\tilde{s}|s)}{P_{\widetilde{S}}(\tilde{s})}\right) - p_d^2 I(S;\widetilde{S})^2 \ . \qquad (78)$$

Proof. To prove the statement we use Theorem 12 and set $S\equiv X$, $\widetilde{S}\equiv Y$ and $(\widetilde{S}_d,\widetilde{S}_a)\equiv Z$. For $\widetilde{S}_d \neq E$ and $\widetilde{S}_a\neq E$ we have that

$$\begin{aligned}P_{S,\widetilde{S},\widetilde{S}_d,\widetilde{S}_a}(s,\tilde{s},\tilde{s},s) &= P_{S,\widetilde{S}}(s,\tilde{s})(1-p_a)\\ P_{S,\widetilde{S}_d,\widetilde{S}_a}(s,\tilde{s},s) &= P_S(s)P_{\widetilde{S}|S}(\tilde{s}|s)(1-p_a)\\ P_{\widetilde{S}|\widetilde{S}_d,\widetilde{S}_a}(\tilde{s}|\tilde{s},s) &= 1 \ , \qquad (79)\end{aligned}$$

for $\widetilde{S}_d = E$ and $\widetilde{S}_a = E$ we have that

$$\begin{aligned}P_{S,\widetilde{S},\widetilde{S}_d,\widetilde{S}_a}(s,\tilde{s},E,E) &= P_{S,\widetilde{S}}(s,\tilde{s})p_d\\ P_{S,\widetilde{S}_d,\widetilde{S}_a}(s,E,E) &= P_S(s)p_d\\ P_{\widetilde{S}|\widetilde{S}_d,\widetilde{S}_a}(\tilde{s}|E,E) &= P_{\widetilde{S}}(\tilde{s}) \qquad (80)\end{aligned}$$

and for $\widetilde{S}_d \neq E$ and $\widetilde{S}_a = E$ it holds that

$$\begin{aligned}P_{S,\widetilde{S},\widetilde{S}_d,\widetilde{S}_a}(s,\tilde{s},\tilde{s},E) &= P_S(s)P_{\widetilde{S}|S}(\tilde{s}|s)(p_a-p_d)\\ P_{S,\widetilde{S}_d,\widetilde{S}_a}(s,\tilde{s},s) &= P_S(s)p_d\\ P_{\widetilde{S}|\widetilde{S}_d,\widetilde{S}_a}(\tilde{s}|\tilde{s},s) &= P_{\widetilde{S}}(\tilde{s}) \qquad (81)\end{aligned}$$

Using Theorem 12 and the identities outlined above directly gives the desired result after some elementary steps. ■

Remark 10. Notice that it is to be assumed that the converse bound is of poor quality as the analog erasure probability p_a has no influence on the bound. The approach taken for the digital attacker to use Theorem 8 to find an upper bound on the maximal rate of the HDA cannot be done for the analog attacker in a straightforward manner. The reason is that Theorem 8 is only applicable if S, $\widetilde{S}$ and $(\widetilde{S}_d,\widetilde{S}_a)$ form a Markov chain, which is not the case.

Table 7: Achievability (ach.) and converse (conv.) results on the number of necessary capacitive PUF cells for $p_d = 0.18$ and $p_a = 0.36$, PUF reliability $\varepsilon = 10^{-6}$ and security levels 128, 192 and 256 bit, analog attacker, equiprobable quantization, $\sigma_P = 2241, \sigma_N = 129$

quantizer	ach.128b	conv.128b	ach.192b	conv.192b	ach.256b	conv.256b
2	1938	902	2923	1295	3909	1683
4	1399	417	2113	607	2828	795
8	1511	239	2219	358	2918	478
16	5983	201	8470	301	10897	401
32	-	187	-	280	-	373
64	-	179	-	269	-	358

Table 8: Achievability (ach.) and converse (conv.) results on the number of necessary capacitive PUF cells for $p = 0.18$ and $p_a = 0.36$, PUF reliability $\varepsilon = 10^{-6}$ and security levels 128, 192 and 256 bit, analog attacker, equidistant quantization, $\sigma_P = 2241, \sigma_N = 129$

quantizer	ach.128b	conv.128b	ach.192b	conv.192b	ach.256b	conv.256b
2	1938	902	2923	1295	3909	1683
4	2305	740	3482	1070	4659	1396
8	1679	363	2537	545	3397	726
16	1355	245	2044	367	2734	490
32	2184	190	3141	284	4081	379
64	-	183	-	275	-	366

The results for applying Theorem 15 and Theorem 16 to the HDA in the analog attacker scenario are presented in Table 7 and Table 8. Empty spots in the tables signal that the amount of required PUF cells is above 20000 and the respective parameter sets therefore have been considered impractical for implementations due to better alternatives. Achievability as well as converse results on the required number of PUF cells are given. Comparing the achievability results to the results for the digital attacker, we observe that for a small amount of quantization levels are almost identical. Even though we cannot say the same for the converse results this is an interesting observation. Our results suggest that it is better to use coarse input quantizers rather than fine ones. Furthermore, for a very low number of quantization intervals equiprobable quantization performs better than equidistant one. We observe that 4 quantization levels with equiprobable input quantization give the best achievability bounds for all security levels in this case. As equiprobable quantization has the benefit of leaking no information about the secret via the reconstruction helper data $\widetilde{W}^n$ (do not confuse this with the quantization helper data W^n) this is particularly interesting. Aside from requiring less PUF cells from an achievability bound perspective quantizers with fewer levels are cheaper and easier to build. Also notice that the output quantizer levels need to be adjusted according to the helper data. In this respect this observation becomes even more important than in the enrollment phase.

9 Applying Converse Results in Security Analysis

In previous wiretap work [GXKF22], the complexity of the attacker has been estimated by $H_{att} = -\sum_i^{n_s} p_{s,i} \log_2(p_{s,i})$, where n_s is the dimension of the code and $p_{s,i}$ denotes the symbol error rate of an attack on the information symbols of the polar code. For the parameters in the paper this leads to a claimed attack complexity of 100 bit. However, going to the physical channel, only np_d symbols are destroyed. For the discussed use case with 128 PUF cells, 23 destroyed nodes and 8 quantization intervals, this leads to a missing information of 69 bits depending on the quantization level. When also considering frozen bits in the polar code and additional quantization leakage, the resulting security level is notably under the claimed 100 bit.

However, there is also another way to show that this scheme cannot achieve the required security level of 100 bit in the digital attacker scenario. We can simply use our converse bound (Theorem 14) and observe that for a security level of 100 bit and error probability 10^{-6} we would require at least 389 PUF cells. Thereby, this construction breaks the converse bound and hence the construction cannot reach the desired security level without substantially leaking information about the secret through the helper data.

10 Summary and Outlook

Over the last years, several practical papers for key generation for PUF-based tamper protection were introduced. In this work, we analyzed the problem with information theoretical tools for additional insights and contribute to understanding and quantifying theoretical and practical limits.

We have given achievable rates under the constraint that the quantization helper data leaks no information about the quantized PUF response S^n obtained during the enrollment phase both in the asymptotic as well as the non-asymptotic setting for two different attacker models. Asymptotically we observed that for the digital attacker equiprobable input quantization performs better than equidistant input quantization in terms of achievable code rates. In case the analog attacker of concern to designer equiprobable quantization can be beneficial for larger quantization alphabets. The same observation holds in the finite length regime. Our results have practical merit since they show that coarse quantization is not only easier to implement but also often requires less capacitive cells on the foil. Furthermore, we presented converse bounds that showed that existing implementations cannot achieve a security level of 100 bits if leakage of information via the public quantization helper data is prevented. The achievability results allow to state an upper bound on the required number of capacitive PUF cells. Thereby, they permit to set design goals for PUFs with respect to the required security and reliability levels.

Our results show that we can achieve the number of required capacitive cells with an optimized implementation, e.g. through a feasible feature shrink. Consequently, by quantifying the fundamental limits of this PUF architecture, the results presented in this work contribute to guiding practical work towards theoretically secure and certifiable future implementations.

Acknowledgements

The authors thank Wei Yang for helpful conversations on his work on wiretap channels in the finite blocklength regime. Further thanks goes to Antonia Wachter-Zeh for helpful discussions. Georg Maringer would like to thank Giuseppe Durisi for hosting him twice for research visits at Chalmers University and in particular for introducing him into the field of finite blocklength information theory on those occasions. The authors especially thank the anonymous reviewers for their comments that significantly improved the quality of this manuscript.

This research presented in this work was supported by the Bavarian Ministry of Economic Affairs, Regional Development and Energy, in the project Trusted Electronics Bavaria Center.

References

[AC93] Rudolf Ahlswede and Imre Csiszár. Common randomness in information theory and cryptography. i. secret sharing. *IEEE Transactions on Information Theory*, 39(4):1121–1132, 1993.

[BB11] Matthieu Bloch and Joao Barros. *Physical-layer security: from information theory to security engineering.* Cambridge University Press, 2011.

[BDH+10] Ileana Buhan, Jeroen Doumen, Pieter Hartel, Qian Tang, and Raymond Veldhuis. Embedding renewable cryptographic keys into noisy data. *International Journal of Information Security*, 9:193–208, 2010.

[BSI24] BSI. TR-02102-1: "Cryptographic Mechanisms: Recommendations and Key Lengths". https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidelines/TG02102/BSI-TR-02102-1.html, 2024. [Version 2024-1].

[BW13] Holger Boche and Rafael F Wyrembelski. Secret key generation using compound sources-optimal key-rates and communication costs. In *SCC 2013; 9th International ITG Conference on Systems, Communication and Coding*, pages 1–6. VDE, 2013.

[CIW+17] Bin Chen, Tanya Ignatenko, Frans M J. Willems, Roel Maes, Erik van der Sluis, and Georgios Selimis. A robust SRAM-PUF key generation scheme based on polar codes. In *IEEE Global Communications Conference (GLOBECOM)*, 2017.

[CK78] Imre Csiszár and Janos Korner. Broadcast channels with confidential messages. *IEEE transactions on information theory*, 24(3):339–348, 1978.

[CT99] Thomas M. Cover and Joy A. Thomas. *Elements of information theory.* John Wiley & Sons, 1999.

[dGSdVL16] Joep de Groot, Boris Skoric, Niels de Vreede, and Jean-Paul Linnartz. Quantization in zero leakage helper data schemes. *EURASIP Journal on Advances in Signal Processing*, 2016(1):54, 2016.

[DGSV15] Jeroen Delvaux, Dawu Gu, Dries Schellekens, and Ingrid Verbauwhede. Helper data algorithms for PUF-based key generation: Overview and analysis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 34(6):889–902, 2015.

[DGV+16] Jeroen Delvaux, Dawu Gu, Ingrid Verbauwhede, Matthias Hiller, and Mandel Yu. Efficient fuzzy extraction of PUF-induced secrets: Theory and applications. In Benedikt Gierlichs and Axel Poschmann, editors, *Conference on Cryptographic Hardware and Embedded Systems (CHES)*, volume 9813 of *LNCS*, pages 412–431. Springer Berlin / Heidelberg, 2016.

[DKM+07] Stark C Draper, Ashish Khisti, Emin Martinian, Anthony Vetro, and Jonathan S Yedidia. Using distributed source coding to secure fingerprint biometrics. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, volume 2, pages II–129. IEEE, 2007.

[DRS04a] Yevgeniy Dodis, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In *International conference on the theory and applications of cryptographic techniques*, pages 523–540. Springer, 2004.

[DRS04b] Yevgeniy Dodis, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In Christian Cachin and Jan L. Camenisch, editors, *Advances in Cryptology (EUROCRYPT)*, volume 3027 of *LNCS*, pages 523–540. Springer Berlin / Heidelberg, 2004.

[EFK$^+$12] Thomas Esbach, Walter Fumy, Olga Kulikovska, Dominik Merli, Dieter Schuster, and Frederic Stumpf. A new security architecture for smartcards utilizing PUFs. In *ISSE 2012 Securing Electronic Business Processes: Highlights of the Information Security Solutions Europe 2012 Conference*, pages 180–194. Springer, 2012.

[FWHP23] Christoph Frisch, Florian Wilde, Thomas Holzner, and Michael Pehl. A practical approach to estimate the min-entropy in PUFs. *Journal of Hardware and Systems Security*, pages 1–9, 2023.

[GGV17] Michael Geis, Karen Gettings, and Michael Vai. Optical physical unclonable function. In *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 1248–1251, 2017.

[GİK15] Onur Günlü, Onurcan İscan, and Gerhard Kramer. Reliable secret key generation from physical unclonable functions under varying environmental conditions. In *2015 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6. IEEE, 2015.

[GİSK19] Onur Günlü, Onurcan İscan, Vladimir Sidorenko, and Gerhard Kramer. Code constructions for physical unclonable functions and biometric secrecy systems. *IEEE Transactions on Information Forensics and Security*, 14(11):2848–2858, 2019.

[GOFK21] Kathrin Garb, Johannes Obermaier, Elischa Ferres, and Martin König. FORTRESS: Fortified tamper-resistant envelope with embedded security sensor. In *International Conference on Privacy, Security and Trust (PST)*, pages 1–12, 2021.

[GS20] Onur Günlü and Rafael F Schaefer. An optimality summary: Secret key agreement with physical unclonable functions. *Entropy*, 23(1):16, 2020.

[GS22] Jan Sebastian Götte and Björn Scheuermann. Can't touch this: Inertial HSMs thwart advanced physical attacks. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, (1):69–93, 2022.

[GSHO21] Kathrin Garb, Marc Schink, Matthias Hiller, and Johannes Obermaier. Attacks and countermeasures for capacitive PUF-based security enclosures. In *IEEE Physical Assurance and Inspection of Electronics (PAINE)*, pages 1–8, 2021.

[GXKF22] Kathrin Garb, Marvin Xhemrishi, Ludwig Kürzinger, and Christoph Frisch. The wiretap channel for capacitive PUF-based security enclosures. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2022(3):165–191, 2022.

[HKS20] Matthias Hiller, Ludwig Kürzinger, and Georg Sigl. Review of error correction for PUFs and evaluation on state-of-the-art FPGAs. *Journal of Cryptographic Engineering*, 2020.

[HO17] Matthias Hiller and Aysun Gurur Önalan. Hiding secrecy leakage in leaky helper data. In Wieland Fischer and Naofumi Homma, editors, *Conference on Cryptographic Hardware and Embedded Systems*, volume 10529 of *LNCS*, pages 601–619. Springer Berlin / Heidelberg, 2017.

[HTW16] Masahito Hayashi, Himanshu Tyagi, and Shun Watanabe. Secret key agreement: General capacity and second-order asymptotics. *IEEE Transactions on Information Theory*, 62(7):3796–3810, 2016.

[HYKD14] Charles Herder, Mandel Yu, Farinaz Koushanfar, and Srinivas Devadas. Physical unclonable functions and applications: A tutorial. *Proceedings of the IEEE*, 102(8):1126–1141, 2014.

[IHKS16] Vincent Immler, Maxim Hennig, Ludwig Kürzinger, and Georg Sigl. Practical aspects of quantization and tamper-sensitivity for physically obfuscated keys. In *Workshop on Cryptography and Security in Computing Systems (CS2)*, pages 13–18. ACM, 2016.

[IHL$^+$18] Vincent Immler, Matthias Hiller, Qinzhi Liu, Andreas Lenz, and Antonia Wachter-Zeh. Variable-length bit mapping and error correcting codes for higher-order alphabet PUFs. *Journal of Hardware and Systems Security (HASS)*, 2(4), 2018.

[IMJFC13] Phil Isaacs, Thomas Morris Jr, Michael J. Fisher, and Keith Cuthbert. Tamper proof, tamper evident encryption technology. In *Pan Pacific Symposium*. SMTA, 2013.

[Imm19] Vincent Immler. *Higher-Order Alphabet Physical Unclonable Functions*. Dissertation, 2019.

[IOK$^+$18] Vincent Immler, Johannes Obermaier, Martin König, Matthias Hiller, and Georg Sigl. B-TREPID: Batteryless tamper-resistant envelope with a PUF and integrity detection. In *IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pages 49–56, 2018.

[ION$^+$19] Vincent Immler, Johannes Obermaier, Kuan Kuan Ng, Fei Xiang Ke, Jin Ju Lee, Yak Peng Lim, Wei Koon Oh, Keng Hoong Wee, and Georg Sigl. Secure physical enclosures from covers with tamper-resistance. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2019(1), 2019.

[IU19] Vincent Immler and Karthik Uppund. New insights to key derivation for tamper-evident physical unclonable functions. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2019(3):30–65, 2019.

[IW09] Tanya Ignatenko and Frans MJ Willems. Biometric systems: Privacy and secrecy aspects. *IEEE Transactions on Information Forensics and security*, 4(4):956–973, 2009.

[LKP09] Yingbin Liang, Gerhard Kramer, and H Vincent Poor. Compound wiretap channels. *EURASIP Journal on Wireless Communications and Networking*, 2009:1–12, 2009.

[Mau93] U.M. Maurer. Secret key agreement by public discussion from common information. *IEEE Transactions on Information Theory*, 39(3):733–742, 1993.

[MGS13] Abhranil Maiti, Vikash Gunreddy, and Patrick Schaumont. A systematic method to evaluate and compare the performance of physical unclonable functions. *Embedded systems design with FPGAs*, pages 245–267, 2013.

[MS77] Florence Jessie MacWilliams and Neil James Alexander Sloane. *The theory of error-correcting codes*. North-Holland, 1977.

[OI18] Johannes Obermaier and Vincent Immler. The past, present, and future of physical security enclosures: From battery-backed monitoring to PUF-based inherent security and beyond. *Journal of Hardware and Systems Security (HASS)*, 2018.

[OIHS18] Johannes Obermaier, Vincent Immler, Matthias Hiller, and Georg Sigl. A measurement system for capacitive PUF-based security enclosures. In *ACM/IEEE Design Automation Conference (DAC)*, 2018.

[RFB$^+$23] Carl Riehm, Christoph Frisch, Florin Burcea, Matthias Hiller, Michael Pehl, and Ralf Brederlow. Structured design and evaluation of a resistor-based PUF robust against PVT-variations. In *International Symposium on Design and Diagnostics of Electronic Circuits and Systems (DDECS)*, pages 93–98. IEEE, 2023.

[SAS17] Taras Stanko, Fitria Nur Andini, and Boris Skoric. Optimized quantization in zero leakage helper data systems. *IEEE Transactions on Information Forensics and Security*, 12(8):1957–1966, 2017.

[SI23] SOG-IS. Joint Interpretation Library – Application of Attack Potential to Hardware Devices with Security Boxes. https://www.sogis.eu/documents/cc/domains/hardware_devices/JIL-Application-of-Attack-Potential-to-Hardware-Devices-with-Security-Boxes-v3.1.pdf, 2023. [Version 3.1].

[SMKT06] Boris Skoric, S. Maubach, Tom A. M. Kevenaar, and Pim Tuyls. Information-theoretic analysis of capacitive physical unclonable functions. *Journal of Applied Physics*, 100(2):024902–024902–11, 2006.

[STZP22] Paul Staat, Johannes Tobisch, Christian Zenger, and Christof Paar. Anti-tamper radio: System-level tamper detection for computing systems. In *IEEE Symposium on Security and Privacy (S&P)*, pages 1722–1736, 2022.

[TSS$^+$06a] Pim Tuyls, Geert-Jan Schrijen, Boris Skoric, Jan Van Geloven, Nynke Verhaegh, and Rob Wolters. Read-proof hardware from protective coatings. In *Cryptographic Hardware and Embedded Systems-CHES 2006: 8th International Workshop, Yokohama, Japan, October 10-13, 2006. Proceedings 8*, pages 369–383. Springer, 2006.

[TSS$^+$06b] Pim Tuyls, Geert-Jan Schrijen, Boris Skoric, Jan van Geloven, Nynke Verhaegh, and Rob Wolters. Read-proof hardware from protective coatings. In Louis Goubin and Mitsuru Matsui, editors, *Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, volume 4249 of *LNCS*, pages 369–383. Springer Berlin Heidelberg, 2006.

[VDRY09] Anthony Vetro, Stark C Draper, Shantanu Rane, and Jonathan Yedidia. Securing biometric data, 2009.

[VNK+15] Michael Vai, Ben Nahill, Josh Kramer, Michael Geis, Dan Utin, David Whelihan, and Roger Khazan. Secure architecture for embedded systems. In *2015 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–5. IEEE, 2015.

[VTO+10] Evgeny A Verbitskiy, Pim Tuyls, Chibuzo Obi, Berry Schoenmakers, and Boris Skoric. Key extraction from general nondiscrete signals. *IEEE Transactions on Information Forensics and Security*, 5(2):269–279, 2010.

[VWN+16] Michael Vai, David J Whelihan, Benjamin R Nahill, Daniil M Utin, Sean R O'Melia, and Roger I Khazan. Secure embedded systems. *Lincoln Laboratory Journal*, 22(1):110–122, 2016.

[Wei00] Steve H Weingart. Physical security devices for computer subsystems: A survey of attacks and defenses. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 302–317. Springer, 2000.

[WGP18] Florian Wilde, Berndt M Gammel, and Michael Pehl. Spatial correlation analysis on physical unclonable functions. *IEEE Transactions on Information Forensics and Security*, 13(6):1468–1480, 2018.

[Wyn75] Aaron D Wyner. The wire-tap channel. *Bell system technical journal*, 54(8):1355–1387, 1975.

[YSP16] Wei Yang, Rafael F Schaefer, and H Vincent Poor. Finite-blocklength bounds for wiretap channels. In *2016 IEEE International Symposium on Information Theory (ISIT)*, pages 3087–3091. IEEE, 2016.

[YSP19] Wei Yang, Rafael F Schaefer, and H Vincent Poor. Wiretap channels: Nonasymptotic fundamental limits. *IEEE Transactions on Information Theory*, 65(7):4069–4093, 2019.

A Basic Notions of Information and Coding Theory

Several notions introduced in this section are standard in information theory. Hence, they can be found in textbooks like [CT99].

Definition 7 (Memoryless channel)**.** We say that a channel is memoryless if it holds that

$$P_{Y^n|X^n}(\mathbf{y}^n|\mathbf{x}^n) = \prod_{i=1}^{n} P_{Y_i|X_i}(y_i|x_i) \ . \tag{82}$$

Definition 8. We say that a channel created an error at the decoder's input if an input symbol x has been mapped to an output $y \neq x \in \mathcal{X}$. We define a symbol erasure to be the event that an input symbol x has been erased and hence this symbol gives the receiver no information about the channel input. Notice that the synchronization between channel input and channel output is not lost in this case as it is when symbols are deleted. If an erasure occurs at the i-th symbol we denote this by setting the random variable $Y_i = E$, where the event E denotes that an erasure occurred.

Definition 9. We denote the average block error probability by

$$\overline{P}_e \coloneqq \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} P_e(m) \ , \tag{83}$$

where

$$P_e(m) = P_{Y^*|\mathcal{M}}(\mathcal{D}(Y^*) \neq m|m) \tag{84}$$

denotes the probability that an error occurs if the message m is transmitted. We denote the maximal error probability by

$$P_{e,max} = \max_{m \in \mathcal{M}} P_e(m) \ . \tag{85}$$

Definition 10. We define the **rate** R of a code of cardinality M and blocklength n to be

$$R := \frac{1}{n} \log_2(M) \ . \tag{86}$$

Definition 11. We define the support of a probability mass function P_X for a random variable X taking values in a set $\mathcal{X}$ by

$$\operatorname{supp}(P_X) := \{x \in \mathcal{X} : P_X(x) > 0\} \ . \tag{87}$$

Definition 12. We define the **entropy** of a discrete RV X to be

$$H(X) := - \sum_{a \in \operatorname{supp}(P_X)} P_X(a) \log_2(P_X(a)) \tag{88}$$

and the conditional entropy of X given that some random variable $Z = z$ to be

$$H(X|Z = z) := - \sum_{a \in \operatorname{supp}(P_{X|Z=z})} P_{X|Z}(a|z) \log_2(P_{X|Z}(a|z)) \ . \tag{89}$$

Furthermore, we denote the conditional entropy of X given Z to be

$$H(X|Z) := - \sum_{(a,b) \in \operatorname{supp}(P_{XZ})} P_{XZ}(a,b) \log_2(P_{X|Z}(a|b)) = \sum_{b \in \operatorname{supp}(Z)} P_Z(b) H(X|Z = b) \ . \tag{90}$$

Definition 13. We define the **differential entropy** of a continuous RV X with probability density function (pdf) f_X by

$$h(X) := - \int_{a \in \operatorname{supp} f_X} f_X(a) \log_2(f_X(a)) \, \mathrm{d}a \ . \tag{91}$$

The definition of the conditional differential entropy of X given some random variable Z is analogous to eq. (90) in Definition 12.

Definition 14. We define the **variational distance** $d(P, Q)$ between two probability mass functions (pmfs) P and Q defined over the same domain $\mathcal{X}$ by

$$d(P, Q) = \frac{1}{2} \sum_{x \in \mathcal{X}} |P(x) - Q(x)| \ . \tag{92}$$

Definition 15. We specify the Kullback-Leibler divergence between two pmfs P and Q over the same domain by

$$D(P||Q) = \sum_{x \in \mathcal{X}} P(x) \log_2 \left(\frac{P(x)}{Q(x)} \right) \ , \tag{93}$$

where we define $0 \log_2(0) := 0$.

Definition 16. The mutual information between two random variables X and Y is specified by

$$\begin{aligned} I(X;Y) &\coloneqq D(P_{XY}||P_X P_Y) \\ &= \sum_{(a,b)\in \operatorname{supp}(P_{XY})} P_{XY}(a,b) \log_2 \left(\frac{P_{XY}(a,b)}{P_X(a)P_Y(b)} \right) = H(Y) - H(Y|X) \ . \end{aligned} \tag{94}$$

If the random variable Y is continuous it holds that

$$I(X;Y) \coloneqq h(Y) - h(Y|X) \ . \tag{95}$$

Definition 17. The mutual information $I(X;Y|Z=c)$ between two random variables X and Y conditioned on a discrete random variable Z taking the value c is defined by

$$I(X;Y|Z=c) := H(X|Z=c) - H(X|Y,Z=c) \tag{96}$$

if X is discrete and

$$I(X;Y|Z=c) := h(X|Z=c) - h(X|Y,Z=c) \tag{97}$$

if X is continuous. Similar to Definition 12 we define

$$I(X;Y|Z) := \sum_{c\in \operatorname{supp}(P_Z)} P_Z(c)\, I(X;Y|Z=c) \ . \tag{98}$$

For continuous Z with pdf f_Z we define

$$I(X;Y|Z) = \int_{c\in \operatorname{supp}(f_Z)} f_Z(c)\, I(X;Y|Z=c)\, \mathrm{d}c \ . \tag{99}$$