High-Resolution EM Attacks Against Leakage-Resilient PRFs Explained

And An Improved Construction

Florian Unterstein[1], Johann Heyszl[1], Fabrizio De Santis[2], Robert Specht[1], and Georg Sigl[3]

[1] Fraunhofer Research Institution AISEC, Munich, Germany
name.surname@aisec.fraunhofer.de
[2] Siemens AG, Corporate Technology, Munich, Germany*
fabrizio.desantis@siemens.com
[3] Technische Universität München, Munich, Germany
sigl@tum.de

Abstract. Achieving side-channel resistance through Leakage Resilience (LR) is highly relevant for embedded devices where requirements of other countermeasures such as e.g. high quality random numbers are hard to guarantee. The main challenge of LR lays in the initialization of a secret pseudorandom state from a long-term key and public input. Leakage-Resilient Pseudo-Random Functions (LR-PRFs) aim at solving this by bounding side-channel leakage to non-exploitable levels through frequent re-keying. Medwed et al. recently presented an improved construction at ASIACRYPT 2016 which uses "unknown-inputs" in addition to limited data complexity and correlated algorithmic noise from parallel S-boxes. However, a subsequent investigation uncovered a vulnerability to high-precision EM analysis on FPGA. In this paper, we follow up on the reasons why such attacks succeed on FPGAs. We find that in addition to the high spatial resolution, it is mainly the *high temporal resolution* which leads to the reduction of algorithmic noise from parallel S-boxes. While spatial resolution is less threatening for smaller technologies than the used FPGA, temporal resolution will likely remain an issue since balancing the timing behavior of signals in the nanosecond range seems infeasible today. Nonetheless, we present an improvement of the ASIACRYPT 2016 construction to effectively protect against EM attacks with such high spatial *and* high temporal resolution. We carefully introduce additional key entropy into the LR-PRF construction to achieve a high remaining security level even when implemented on FPGAs. With this improvement, we finally achieve side-channel secure LR-PRFs in a practical and simple way under verifiable empirical assumptions.

Keywords: Leakage-resilient cryptography, PRF, high-resolution localized EM attacks, AES

* The work was conducted while the author was with Technische Universität München.

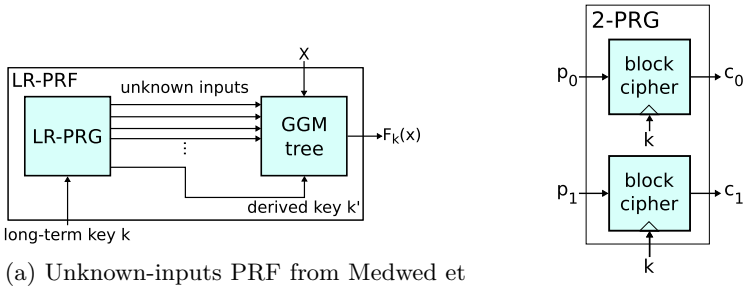
1 Introduction

Even though the contribution of the paper extends beyond the application on FPGAs, FPGA security has been our main motivation. FPGAs and especially System-on-Chips (SoCs), that integrate powerful embedded CPUs and FPGAs on the same chip, are currently being designed into application domains such as automotive, industrial control systems and defense. A secure startup in the field is crucial for devices of said domains, especially since adversaries may be able to perform side-channel measurements and may even repeatedly reboot the device. In this context, our research goal is a side-channel protected cryptographic engine which can e.g. be used to securely decrypt and authenticate firmware images and FPGA configurations during startup or remote updates.

Protecting cryptographic engines on FPGAs against side-channel analysis, however, is challenging. Conventional approaches to protect block cipher implementations are masking of secret intermediate values [4] or reducing the observable Signal-to-Noise Ratio (SNR) by time-based shuffling [11]. However, such techniques require fresh true randomness which is difficult to satisfy in practice. Contrary to security controllers, which use internal clocks for de-synchronization, which is another form of time-based hiding, FPGAs are usually clocked from outside of the FPGA, hence, side-channel measurements are always perfectly synchronized. Other countermeasures on the logic level such as e.g. dual-rail logic styles [9] come with significant implementation overheads and have recently been shown to be ineffective on FPGAs [8].

Leakage-resilient symmetric constructions, in contrast, wrap block ciphers in a mode of operation, which is inherently resilient against side-channel attacks by bounding the exploitable leakage through frequent key changes. In this respect, they represent a significant value to the security of FPGA implementations because no randomness is required to effectively protect against powerful DPA attacks. In order to meet application needs, such constructions are typically *stateless*, i.e. no additional secret synchronization values are available. This means that re-keying constructions still use a constant secret key and public input at the start of their operation. The protection of this initialization phase is most challenging because repeating side-channel measurements of it cannot be limited for attackers — this is called the secure initialization problem in this context.

Leakage-Resilient Pseudo-Random Functions (LR-PRFs) based on the tree construction of Goldreich, Goldwasser and Micali [5] (GGM tree) were proposed to solve this issue. They bound the observable *data complexity* for an attacker to a minimum, i.e. the attacker can only observe two different plaintext values per key. The *measurement complexity* is still unlimited, which means that attackers may repeat the two operations and average the measured traces for noise reduction. Medwed et al. [12] describe an AES-like block cipher in this context where all S-boxes are used in parallel with equal inputs and have the same leakage function (the so-called equal leakage assumption). The parallel S-boxes lead to correlated algorithmic noise which hinders attacks on single key parts. They show that using equal inputs to the S-boxes allows a trade-off between security and efficiency by



(a) Unknown-inputs PRF from Medwed et al. [13]

(b) 2-PRG

Fig. 1: LR-PRF and underlying 2-PRG building block

increasing the data-complexity to values of 4 and greater. They also state, that the number of parallel S-boxes to achieve at least 80 bit security is 24, which means that AES with its 16 S-boxes would not be a suitable candidate. However, Belaïd et al. [1] later showed that such alternative cipher designs with sufficient parallelism can still be broken by high-precision EM attacks when the data complexity is ≥ 16. Recently, Medwed et al. [13] proposed a new idea to achieve an efficient GGM tree PRF based on the AES block cipher, by using *unknown-inputs* which is depicted in Fig. 1a. The unknown-inputs are generated once by a Leakage-Resilient Pseudo-Random Generator (LR-PRG) [15] which is built using a block cipher with fixed inputs and, crucially, with the minimum possible data-complexity of 2 (implementing the functionality of a length-doubling 2-PRG as shown in Fig. 1b). Each iteration of the LR-PRG evaluates the 2-PRG, outputs c_1 and uses c_0 as key for the next iteration. This LR-PRG uses the long-term key k also to derive the initial key k' for the GGM tree (right part in in Fig. 1a), which, like the unknown inputs, directly depends on k and, thus, represents no additional fresh key entropy. Using the generated unknown inputs, Medwed et al. [13] argue that the main GGM tree can be based on a regular AES block cipher with 16 parallel S-boxes while supporting the same efficiency as their earlier proposal [12]. However, Unterstein et al. [17] showed that also this construction (precisely, the LR-PRG part in Fig. 1a) with data complexity of 2 can be attacked using multivariate localized EM attacks.

Contributions As a first contribution, we investigate the reasons why AES-based leakage-resilient constructions with 2-limited inputs can be attacked with high-resolution localized EM analysis on FPGAs through a careful laboratory evaluation. Our results clearly show that the isolation of the S-boxes *mainly* occurs in the time domain and not, as currently believed, in the spatial domain. Hence, the major key entropy loss is due to the *high temporal resolution* of localized EM measurements, whereas the high spatial resolution still plays a fundamental role but could be partially mitigated by hand-crafted place&route (P&R) constraints. Also, we confirm that low-precision current side-channel measure-

ments are not able to distinguish the individual key bytes of LR-PRFs in practice, due to inherent low-pass filtering effects.

As a second contribution we show through simulations that contrary to the equal-leakage assumption of S-boxes in Medwed et al. [12], S-boxes may in fact exhibit unequal leakage characteristics as long as the allowed data-complexity is very low, e.g. limited to 2. This means designers may simply aim for an overall low area and disregard designing equally leaking S-boxes which should require less design effort and at the same time be more robust to spatial separation.

We believe that it is unrealistic to prevent considerable entropy loss through localized EM attacks as shown here and by Unterstein et al. [17] in practical scenarios. For example, it seems practically infeasible to craft P&R constraints to achieve synchronous timings for all S-boxes to mitigate temporal separation vulnerabilities[4]. Hence, as a third and most important contribution, we propose to modify Medwed et al.'s [13] unknown-inputs LR-PRF to use *additional key entropy* in the form of one or more additional long-term keys to cope with the inevitable entropy loss in the first part (LR-PRG in Fig. 1a) of their construction. As a result, we obtain a LR-PRF construction where the number of additional long-term keys used in successive 2-PRG iterations depends on the encountered loss of entropy and is a trade-off between security and overhead. We believe that our improved construction can provide a practical and simple solution to the initialization problem of LR-PRFs in face of state-of-the-art multivariate attacks using high-resolution localized EM measurements and under verifiable empirical assumptions. Hence through this improvement we finally achieve a protected engine for practical FPGA applications at the rewarding expense of requiring longer keys.

Organization First, we give preliminary information about the measurement setups and analysis in Section 2. In Section 3 we present a detailed investigation of the reasons why localized EM attacks are successful against state-of-the-art unknown-inputs LR-PRF implementations on FPGAs. Section 4 re-visits the equal-leakage assumption to evaluate its relevance for the case of limited data complexity using simulated template attacks. In Section 5, we sum up and discuss the results of the analysis and draw conclusions for the following Section 6, where we describe our improved unknown-inputs LR-PRF construction.

2 Preliminaries

In this section, we provide background information on the design under test, measurement setup, and evaluation techniques used in the remainder of this paper.

[4] This is particular true on FPGA platforms where there is no control over the physical design of the underlying nanotechnology.

Design Under Test We use an AES-128 hardware design with 16 parallel Canright S-boxes [3] in the datapath and 4 additional S-boxes in the key schedule which are operating at the same time. Only S-boxes synthesized from logic gates allow the required placement flexibility, contrary to RAM-based S-Box designs, and Canright's proposal is state of the art. The S-boxes were specifically designed as hard-macros making them completely equal in terms of cells and routing within the S-box. This is to fulfill the equal leakage assumption stated by Medwed et al. [12]. (We only found out later, that the equal leakage assumption may be disregarded. The details of which can be found in Sect. 4.) We placed them close to each other in the attempt to make the routing, e.g. to the mix-columns logic, as similar as possible, so that the leakage overlaps. We implemented the design in a Xilinx Spartan 6 XC6SLX9-3TQG144C FPGA manufactured in a 45 nm process technology. Estimated from the reports of the design tool (Xilinx ISE 14.7), the die area occupied by the entire AES is about 0.5 mm^2 which is large compared to the size of the probe.

Measurement Setup We use a Langer ICR HH 100-27, 100 µm diameter EM probe and stepping table similar to [17]. A LeCroy WavePro 725Zi oscilloscope with 2.5 GHz bandwidth and a sampling rate of 5 GS/s is used. The test device is clocked at 20 MHz. We take measurements in a 40×40 grid, which results in a step size of 70 µm, on the surface of the decapsulated FPGA. Within each trace, we concentrate on the time duration where the first round S-box look-up is computed. We also perform current measurements using a LeCroy AP033 active differential probe over a 10 Ω shunt resistor in the supply line. We removed capacitances from the FPGA board to reduce the low-pass filtering of the power consumption to gain as much information as possible. We measured 10.000 traces per location for the grid scan, an additional 650.000 traces for each of the 16 S-boxes at their respective locations and 1.000.000 power traces. All measurements were taken using random inputs to the AES.

SNR and Correlated Algorithmic Noise In case of EM measurements, the location has a high influence on the quality of the analysis. Therefore, different measurement locations are usually selected for different targeted signals, i.e. S-boxes in our case. The selection of Locations Of Interest (LOIs) can be done based on different metrics. We select LOIs for different S-boxes by looking for highest mean Signal-to-Noise Ratio (SNR) over time of these S-boxes[5].

We use the common definition of the SNR [10] to quantify the exploitable signal. To compute the SNR over time (SNR trace) of one individual S-box in a measurement, we partition the traces according to the input values of this S-box b and compute its SNR with the estimated mean trace μ_i^b and variance trace $\sigma_i^{2\,b}$ over all traces with input value i at this S-box as:

[5] The selection of LOIs could possibly be improved by using a different metric, however, this will not affect the main findings of this contribution.

$$SNR^b = \frac{Var(Signal^b)}{Var(Noise^b)} = \frac{Var(\mu_0^b, \ldots, \mu_{255}^b)}{Mean(\sigma_0^{2\,b}, \ldots, \sigma_{255}^{2\,b})}. \quad (1)$$

To estimate the signal strength of every individual S-box, we use measurements, where the input data is random and the data complexity is not limited. This means that when computing the SNR as described above, the signal from "other" S-boxes will contribute as uncorrelated algorithmic noise. This leads to SNR values without the correlated algorithmic noise which the targeted constructions leverage upon and allows us to improve our understanding of the relative proportions of the contained signals.

During an actual execution of e.g. a LR-PRG, the data complexity of the construction is, contrarily, limited to two and all plaintext input bytes are equal (carefully chosen inputs). This leads to correlated noise from the other S-boxes, which is persistent after averaging since the respective plaintext inputs are not independent and random anymore. The measurement- and electric noise is still averaged-out in the limited case. Signals of all other S-boxes, estimated through their respective SNR, will cause noise and decrease the exploitable effective SNR accordingly. How exactly this affects an attack is highly dependent on the concrete value of the key and the two plaintexts. At one extreme instance, for an unlucky combination of key and plaintexts, all non-targeted signals might sum up to the same value for both plaintexts, thus not affecting the attack at all. In another instance, the difference of their sums might be large enough to hide the changes of the targeted signal.

Unfortunately, directly calculating the SNR of limited data complexity and this correlated noise is not feasible. To estimate the variance of the signal, i.e. the mean traces for each S-box input value, all other S-boxes would need to be considered since they are correlated, which would require the calculation of 2^{128} mean and variance traces. However, it is intuitively clear that the lower the combined signal strength of the other S-boxes is, the higher is the chance to recover the targeted signal parts. Hence, the SNR without correlated noise is an informative indicator for the expected success rate of an attack on individual S-boxes.

LDA Linear Discriminant Analysis (LDA) is a well-established statistical method to transform high-dimensional data into a lower-dimensional subspace by using the class labels to maximize class separability. In the context of side-channel analysis, the classes correspond to all possible S-box input/output values which are targeted during attacks. This means that an LDA transformation is always done with respect to the signals of one particular S-box which are to be distinguished. In the following analysis, we sometimes show the SNR of LDA-transformed traces instead of the original traces to condense the available and exploitable signal in few dimensions. This increases the ability to visually compare SNR from different S-boxes, but really only helps visual inspection. A profiled attack does perform equally well before and after the LDA transformation [2].

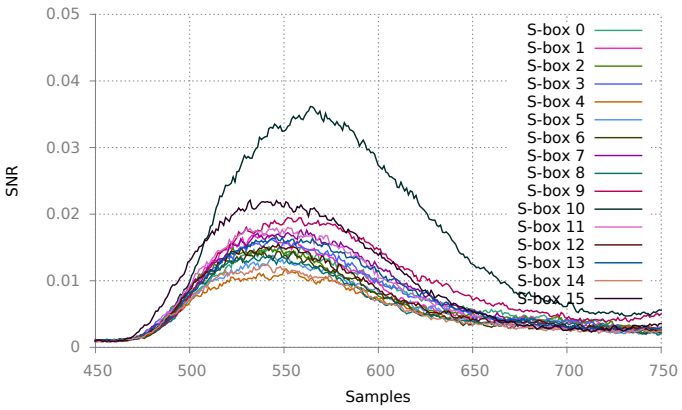
3 Understanding why Localized EM Attacks break Leakage-Resilient PRFs

Recent results from Unterstein et al. [17] showed that multivariate localized EM attacks on FPGA implementations of state-of-the-art leakage-resilient constructions based on AES can reduce the key entropy to levels which are computationally feasible. In this analysis, we use a similar setup as in [17] to investigate the causes for this in detail. We find that beyond the location dependence, which helps to isolate the leakages of single S-boxes, the signals from individual S-boxes get also very well isolated by the high temporal resolution of the measurement setup.

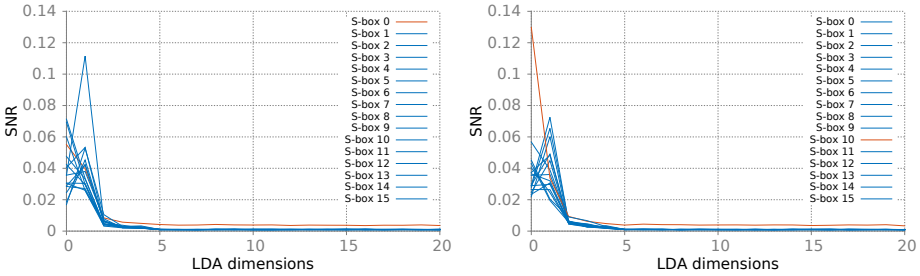
3.1 Leakage-Resilience Holds with Current Measurements

Before analyzing the reasons why EM measurements break resilient constructions, we look at the case of current measurements. For such measurements, Unterstein et al. [17] reported that profiled multivariate DPA is not successful. This means that the algorithmic noise from the parallel hardware and the limited data complexity works as intended. In other words, the algorithmic noise from the respective other S-boxes makes attacks on individual S-boxes infeasible.

Fig. 2a shows the SNR traces of all 16 S-boxes around the time the first round S-box function is computed. The duration of one clock cycle is 250 samples, the positive edge of the clock approximately occurs at sample number 460. It can be seen that the signals of all S-boxes stretch over a time period which is almost the entire clock cycle. This is expected with such kinds of measurements due to the high amount of parasitic capacitances and inductivities which low-pass filter the signal. Most importantly, we note that the signals of the different S-boxes are very uniform in shape and amplitude and that the maximum SNR values of about 0.04 are relatively low. The fact that all S-boxes emit their signal at about the same time means that every S-box will effectively produce noise for every other S-box, thus, leading to the generally low SNR. This is exactly what the construction was meant to achieve. Furthermore, we inspect the SNR after LDA transformation for two cases. One case is S-box number 10 which seems to result in the highest SNR values as observed from Fig. 2a. The other case is S-box number 0, which is one of the S-boxes from the group that exhibits similar SNR values. Fig. 2b and 2c depict the SNR values after LDA for these two cases. The signal of the targeted S-box is plotted in red, while the signals of the other S-boxes are plotted in blue. It is important to note that for each individual figure, first the LDA transformation that fits the signal of the targeted S-box is calculated and applied to the traces. Then the SNR of all S-boxes is calculated in that subspace. The case of S-box 0 in Fig. 2b is representative of the most frequent situation and shows that the signals of all S-boxes are in a similar range. Hence, the targeted signal of S-box 0 is similar or even lower than the signals of the other S-boxes which produce noise. This explains why attacks in such cases are unsuccessful, i.e. the algorithmic noise works as intended. Even the single best case of S-box 10 in Fig. 2c shows that the signals of the other



(a) SNR in current measurement



(b) SNR after LDA transformation in subspace calculated for S-box 0 in red. Others blue.

(c) SNR after LDA transformation in subspace calculated for S-box 10 in red. Others blue.

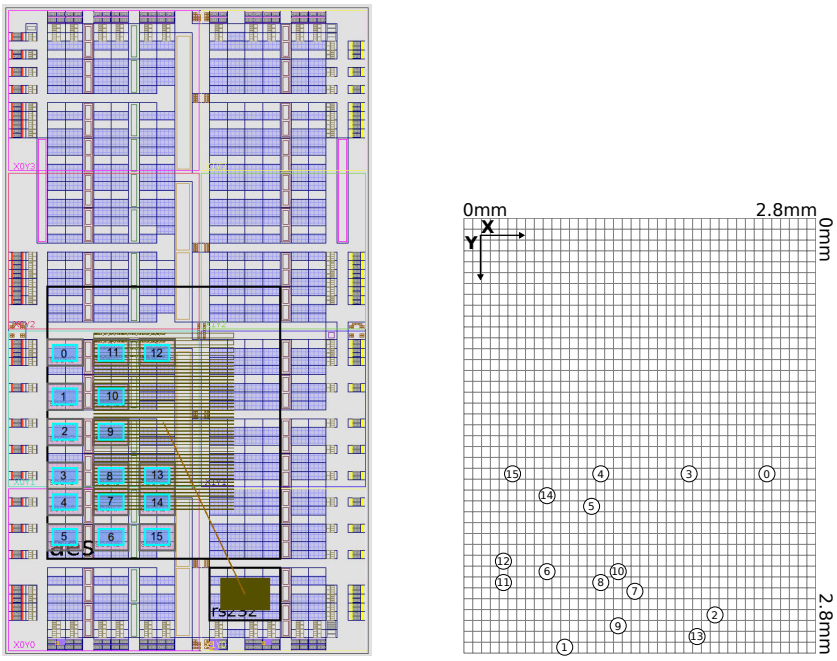
Fig. 2: SNR of S-boxes before and after LDA transformation

S-boxes are relatively high (at approximately $\frac{1}{2}$ to $\frac{1}{3}$ of S-box 10) which also leads to significant noise for this best case.

3.2 Leakage-Resilience Fails when EM Measurements Resolve Signals with High Spatial and Temporal Resolution

The goal of this section is to explain why the parallelism of S-boxes for leakage resilience fails when using localized EM measurements. A natural assumption is that the high-precision setup would lead to measurements where, at the location of a specific S-box, only this S-box exhibits a high SNR while all others exhibit negligible SNR. We show that this is rarely the case, therefore we need another explanation.

We performed EM measurements, selected the LOIs for each S-box, and computed SNRs as described previously. Fig. 3 shows the physical placement locations of the S-boxes on the FPGA floorplan in Fig. 3a and the measurement locations of the same S-boxes (LOIs) in Fig. 3b. The measurement positions are

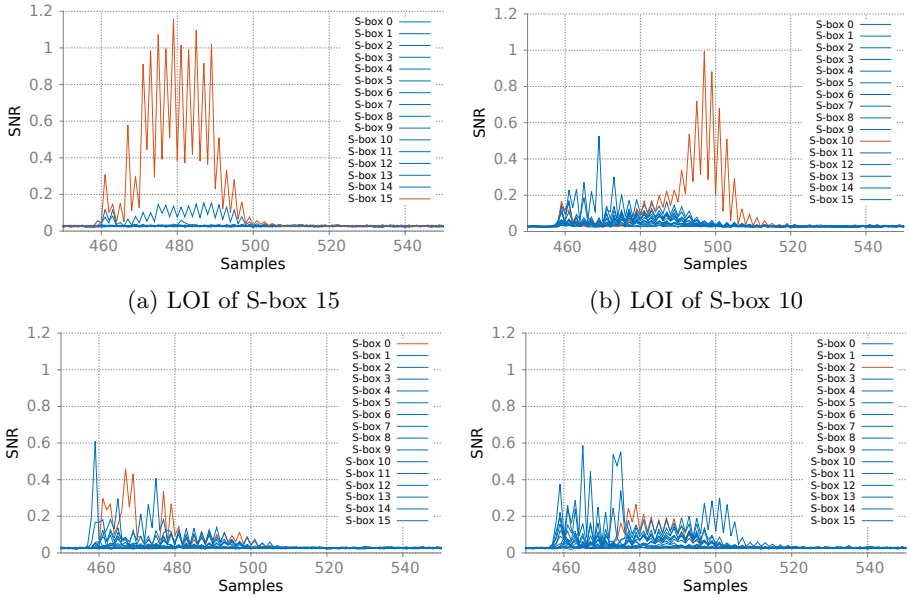


(a) Placement on FPGA (b) Measurement locations (LOIs)

Fig. 3: Placement of S-boxes compared to resulting measurement locations

shown as a grid within a quadratic area of about 7.8 mm^2 in between the bonding wires of the decapsulated FPGA. It is hard to match the two areas exactly, but the measurement grid (right) covers only a part of the floorplan (left) since the positioning of the probe is limited by the bonding wires. What is interesting while comparing the two figures is that apart from a general similarity that all S-boxes are situated and measured on the lower left, there is no reasonable placement-to-measurement correspondence. This already hints that we measure signals at the LOIs which have already propagated through the circuit from their origin in the S-box, e.g. through the power grid.

Fig. 4 depicts the SNRs of all S-boxes at four LOIs, which have been selected to be best for S-boxes 15, 10, 0, and 2. The four shown cases are representative of the 16 LOIs in total which are given in Section 8. The figures each show the SNR of the targeted S-box in red and the SNR of all other S-boxes in blue. As a first observation it should be noted that all detectable signals extend over a significantly shorter time period compared to the power analysis. Specifically, they extend over about 50 time samples which corresponds to a time span of 10 ns. This is short compared to the clock cycle duration of 50 ns (250 samples). In fact, it is close to the critical path delay of 15 ns reported by the synthesis tool. This is similar to the findings of Heyszl et al. [7] and confirms that there are only a few parasitics in the measurement chain.



(a) LOI of S-box 15

(b) LOI of S-box 10

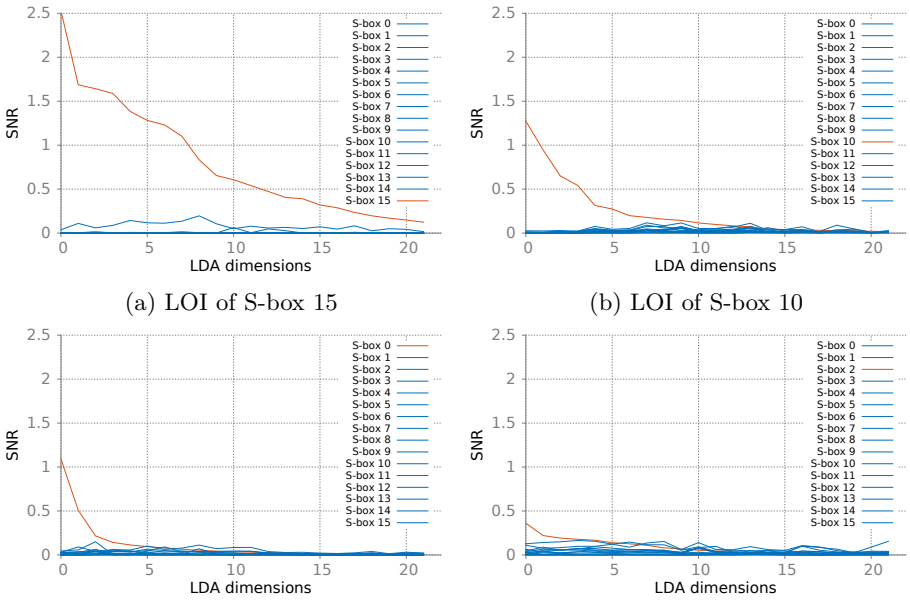
(c) LOI of S-box 0

(d) LOI of S-box 2

Fig. 4: SNRs at four LOIs of targeted S-boxes (red). Others in blue.

As an important observation, the SNR values in Fig. 4 are very high compared to the results from the current measurement. Fig. 4a depicts the situation of S-box 15 which confirms the assumption, that an isolation of S-box signals can, in cases, be achieved through location-dependence. The SNR of S-box 15 is high while the SNR of the other S-boxes is minimal. The case of S-box 10 in Fig. 4b is very different. The SNR of this S-box is again isolated, but only at a certain and precise time. There are times, respectively sample points, where the SNR of other S-boxes is also significant. But at the time samples where the SNR of S-box 10 is highest the others tend to zero. Fig. 4c and 4d depict more cases where there is a strong overlap of signals from different S-boxes. However, again, at certain time-samples the SNR of other S-boxes is small compared to the SNR of the targeted S-box.

In order to make visual inspection easier, we provide the SNR after LDA in Fig. 5. It can generally be noted how LDA compresses the available SNR into the highest dimensions. Unsurprisingly, in cases where the separation, in terms of relative proportion of targeted signal to the other signals, before the LDA transformation has already been high, this becomes significantly more visible after LDA. Fig. 5a depicts S-box 15 and Fig. 5b depicts S-box 10. The high SNR values of the targeted S-boxes, 2.5 and 1.3, and very low SNR values of the other S-boxes in the first dimensions are significant and lead to the assumption that attacks on these S-boxes will succeed with very high probabilities. However, also for S-box 0 in Fig. 5c the proportion of its signal to other signals seems



(a) LOI of S-box 15

(b) LOI of S-box 10

(c) LOI of S-box 0

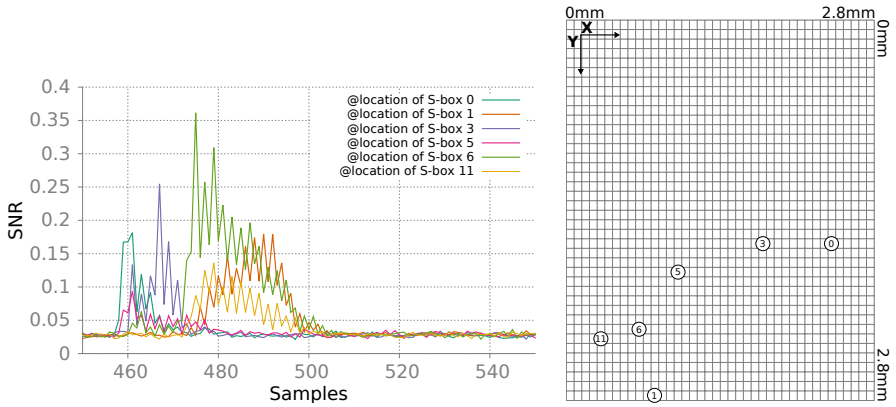
(d) LOI of S-box 2

Fig. 5: SNRs after LDA at four LOIs of targeted S-boxes (red). Others in blue

exploitable in this view, despite the overlap in the time domain. Even for the case of S-box 2 in Fig. 5d the LDA-transformed SNR hints that there is exploitable SNR.

SNR over Location and Time As an example, we analyze the SNR of one particular S-box, number 6, at different measurement locations. Specifically, we simply used several LOIs of other S-boxes. The selected positions are depicted in Fig. 6b. The SNR of S-box 6 at those positions is shown in Fig. 6a. It can be observed, that the SNR crosses a significant threshold at all positions. Depending on the location, though, it appears in different amplitude and different shape over time.

Discussion The most important observation from our analysis is that the leakage signals of different S-boxes are very different when observed with high-precision, low-parasitic EM measurements. This difference is especially remarkable since the S-boxes were carefully designed with equal internal structure and routing. The leakage signal is in fact detectable at different time samples within a very short time range. To the best of our knowledge, the reasons are within-circuit signal propagation delays, or race-conditions. Hence, depending on circuit differences and depending on the position of the measurement relative to the source of the signals which propagate through the circuit, the timing of dif-



(a) SNR of S-box 6 at different locations (b) Measurement locations

Fig. 6: SNR of S-box 6 at different locations

ferent S-boxes is different. As an important insight, we derive that a successful isolation of S-box signals is partly due to the timing of their propagation over the circuit. Hence, the success of attacks on parallel constructions should not be solely attributed to spatial isolation. In fact, a combination of spatial and temporal separation leads to exploitable leakage with distribution over time being dependent on the measurement location.

4 Do We Need Equally Leaking S-Boxes?

Our EM measurements clearly showed that the leakage of the S-boxes is separable because their signals do not overlap enough to generate effective algorithmic noise. One way to increase signal overlap is to place the design closer together. The minimum area occupied by the AES in the evaluated design is determined by the hard-macro placement of the S-boxes, which was originally used with the intent to achieve similar leakage functions. If the individual S-boxes were placed without this constraint, they could be placed interleaved and packed much tighter, but this would inevitably violate the equal-leakage design paradigm.

Previous contributions on the carefully-chosen input LR-PRF [12] as well as on the unknown-inputs LR-PRF [13] also argue the security based on this equal-leakage assumption. Hence, our question is, whether equal leakage is really required in this context or if we can sacrifice it in exchange for tighter, interleaved placement. In this section, we show that S-boxes do not necessarily need to have equal leakage characteristics when the data complexity is low.

Medwed et al. [13] simulated a profiled univariate template attack on noise free traces where the leakage of each S-box is exactly the Hamming weight of its output. The leakage trace, i.e. sample, since it has only a single point, is the sum of the leakage of all S-boxes. This setting represents the worst case for an attacker since the signals from all 16 S-boxes perfectly overlap, hence, produce noise. We modified their simulation by using different probability mass functions for the leakage of the different S-boxes, which we individually randomized such that they deviate from the Hamming weight leakage. Similar to the Hamming weight model, we assume that the total leakage of the S-box is the sum of the contribution of all the bits of the output value. However, for each S-box and each of its output bits, we drew the value from a discrete normal distribution $\mathcal{N}(100, \sigma^2)$. We increased the leakage's codomain so that all distributions and calculations can remain discrete, otherwise the computational cost would become prohibitive. This model is realistic in the sense that we expect the leakage to be somehow dependent on the bit values, albeit some bits will have a stronger and different impact than others. As a corner case, we also performed an (unrealistic) simulation where we randomly assigned leakage values to S-box output values.

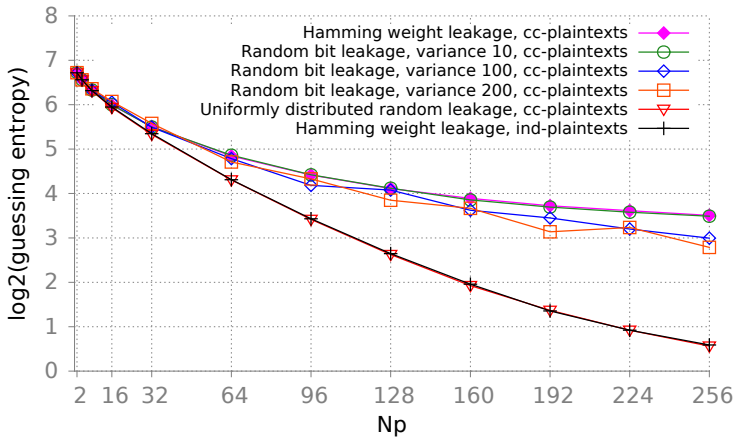


Fig. 7: Remaining guessing entropy after simulated attacks on one key byte with different leakage models (cc=carefully-chosen, ind=independent)

Fig. 7 depicts the guessing entropy of one key byte after such a simulated template attack in relation to the number of observable plaintexts N_p for different simulations. We performed 100.000 simulations per data point and averaged the guessing entropy. For comparison, we included the equal-leakage Hamming-weight model with both carefully-chosen plaintexts, where all bytes are equal, and randomly-chosen plaintexts with independent bytes. It can be seen, that the curve of the equal-leakage Hamming-weight model forms the upper boundary of the guessing entropy; this is the best we can expect. In general, the guessing

entropy goes down with the number of different plaintexts that an attacker can observe. If we randomize the bit leakage, i.e. make the leakage behavior increasingly dissimilar, then the guessing entropy reduces at a faster rate.While the difference for variance 10 is marginal, for variance 100 and 200 this effect becomes obvious. The extreme case of this is the uniformly distributed random leakage, which is in line with the curve of the Hamming weight model with randomly-chosen plaintexts. This is expected because if either the S-box input or the resulting leakage is random, then there can be no correlation between the leakage of S-boxes and, thus, no correlated algorithmic noise. That is the best case for an attacker and leads to the lowest guessing entropy. We can observe and conclude that, for very low data complexities (e.g. 2 or 4), the leakage model does not make a difference on the security of parallel constructions.

5 Summary of the Main Findings and Recommendations

Our experiments and analyses clearly show that state-of-the art EM measurement equipment is able to separate signal contributions of individual S-boxes from parallel FPGA implementations. We investigated the reason for this and derive that the combination of spatial and temporal separation leads to exploitable leakage.

For parallelism to work in the intended way, the S-boxes' leakage should be small and not separable in the time or space domain to achieve security against localized EM attacks. This is typically very hard to achieve on FPGAs because of the limited influence of the hardware designer due to the immutable internal structure of the building blocks and the restricted routing options. While further investigation in this direction seems possible, we are pessimistic about its benefit. We suspect similar issues even on recent 16 nm FPGAs which allow placing the design into a smaller overall area. The reason for this is, that even if S-boxes are placed in a much smaller area with such technologies, and one could argue that a location-dependent isolation may be impossible, the timing of signals of different S-boxes may still be different, allowing an isolation of said signals over time.

On a more optimistic note, we found that with limited data complexity it does not matter if the leakage behavior of the S-boxes is equal. This gives hardware designers more freedom when placing the design since no effort has to be made to craft S-boxes with similar leakage functions. Hence, as a design recommendation we state: parallel S-boxes should be concentrated and densely packed, while interleaving the S-boxes with no regard for their individual layout. In this way the signals of at least a subset of the S-boxes should overlap and cause as much algorithmic noise as possible. This should be sufficient to reach acceptable security levels for this part of the construction so that the improvement presented in the next section can leverage on this to achieve a high overall security level. Nevertheless, it seems unavoidable to perform practical investigations, such as the ones described here, to ensure that the algorithmic noise is effective.

6 Unknown-Inputs Leakage-Resilient PRF with Improved Resistance against Localized EM Attacks

In order to improve existing leakage-resilient PRFs with respect to localized EM attacks, one can either try to prevent the loss of entropy with higher physical design efforts (placement, routing and timing constraints) or compensate it by adding extra key-material. As argued before, it seems hard to design a device in which all S-boxes leak perfectly synchronous and where S-boxes cannot be separated spatially.

Instead, we propose to modify the construction from Medwed et al. [13] in Fig. 1a so that additional key entropy is added to compensate the entropy loss when the construction is subjected to localized EM attacks. We specifically propose to use their construction with two or more long-term keys instead of one, depending on the amount of entropy loss. The first one is used in the generation of the unknown inputs by the 2-PRG as before, the second, additional, one for the subsequent GGM stage itself. This concept can be generalized to use multiple stages of the 2-PRG to further increase the entropy. In that case, another new key is introduced with each such stage.

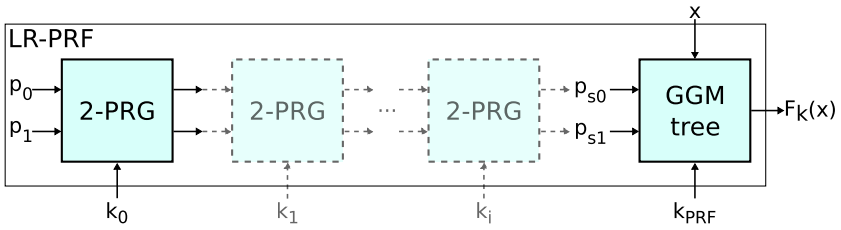


Fig. 8: Improved leakage-resilient PRF construction, dashed parts are optional

More formally, we construct a leakage-resilient PRF $F_k(x)=y$ with $k=(k_0,\ldots,k_i,k_{PRF})$ where $i \geq 0$. Consequently, the minimum required key length with $i=0$ is 256 bits in case of AES-128. Our proposed modified construction is depicted in Fig. 8.

The initial 2-PRG stage uses *known* inputs since using unknown inputs is not possible at this stage. Two encryptions are performed under key k_0 with known plaintexts p_0 and p_1 (usually valued '0' and '1'), and ciphertexts c_0 and c_1 are retrieved (Fig. 1b). This is the part of the construction, where due to the reasons explained in this paper, parts of the key entropy will be lost inevitably. Depending on the quality of the implementation, hence, the amount of lost entropy, we then use c_0 and c_1 as either: (1) *unknown* plaintexts in subsequent iterations of the same 2-PRG stage, while each time introducing a new key $k_1,\ldots,k_i$ to further increase the entropy, or (2), as the *unknown* inputs p_{s0} and p_{s1} to the leakage-resilient PRF, the GGM tree. This GGM tree construction is standard with two possible branches in each stage and 128 iterations to process a public input x with 128 bits.

The idea is that the remaining key entropy of the first 2-PRG stage, which is contained in c_0 or c_1, carries over to the subsequent unknown-input GGM stage and hinders an attack on k_{PRF} in the way argued by Medwed at al. [13] who describe that this would require second order attacks and that there is no straightforward way of testing key candidates. A potential attacker would first need to learn p_{s0} and p_{s1} before he could launch an attack on k_{PRF} using first order DPA. A similar reasoning applies to all potential 2-PRG stages which use unknown inputs as well. How many stages and keys are needed depends on the leakage of the circuit and has to be evaluated through laboratory analysis. Conveniently, this number of necessary repetitions of the 2-PRG stage can for instance be a matter of configuration after the evaluation of a concrete hardware implementation. We expect that for many designs (as the one we analyzed here) one 2-PRG stage is sufficient. However, we like to emphasize that the overhead of adding stages lies solely in key memory and execution time. The entire construction can be implemented using only a single AES core for the GGM tree and the 2-PRG stages.

Note that this construction does not allow to increase the data complexity of the GGM tree levels for more efficiency. The reason is that the generation of more than two unknown plaintexts is not possible without losing additional entropy. Consider the LR-PRG used in the original unknown-inputs LR-PRF proposal which iterates the 2-PRG multiple times, using c_0 as key for the next iteration and returning c_1 as output. Since the plaintext inputs are always known, attacks can be launched on *every iteration* and the resulting key candidate lists can be easily matched across the stages. Because of this, we accept limited efficiency in exchange for improved security.

6.1 Security Discussion

The security of the construction can be discussed along two major attack paths which connect in the middle:

1. The first attack path targets the 2-PRG with known inputs, which is the first part in Fig. 8. This is where we provided a crucial improvement to cope with the loss of entropy due to our findings, and explain how this additional key entropy increases the overall security level.
2. The second attack path targets the GGM tree in Fig. 8, or more generally, all 2-PRGs with unknown inputs within this tree, as well as in the optional part in Fig. 8. Regarding this part, we revisit the argumentation of Medwed et al. [13], and argue that a recent attack on secret inputs from Unterluggauer et al. [16] can be reduced to the same case.

Part 1: Mitigating the Loss of Entropy in the 2-PRG An attack on the first part, i.e. the 2-PRG with known inputs, has been shown to reduce the key entropy of k_0 in Fig. 8 to lower levels than previously expected. Generally, the remaining key entropy of k_0 in Fig. 8 can be denoted as 2^e. In the example provided by Unterstein et al. [17], this amounted to $\approx 2^{50}$ which is within

practical enumeration levels. Hence, it required an improvement because k_0 had been the single source of long-term key entropy.

With our tweak, the first 2-PRG with *known inputs* is followed by one or more 2-PRG stages with *unknown inputs and additional key entropy*. The case with the minimum number of two such stages arises, when the 2-PRG is followed directly by the GGM tree. Then the first level of the unknown-inputs GGM tree can be seen as a separate 2-PRG [14] with unknown inputs and new key k_{PRF}. The subsequent tree levels are a concatenation of 2-PRGs with different keys, which are, however, all depending on k_{PRF} and thus add no entropy.

After this modification, an attacker has no way of verifying key candidates resulting from an attack on the known inputs 2-PRG since the outputs are not accessible. Instead, he must continue along the chain of 2-PRGs which we described above where he encounters new key entropy (at least additional 128 bit). Contrary to the first 2-PRG, all later 2-PRGs operate on unknown inputs. A valid strategy for an attacker is to test resulting candidates for k_0 out of the 2^e pool, and calculate the outputs of the 2-PRG to use them as hypothetical inputs to perform the same attack on the next 2-PRG. This attack on the next iteration has to be repeated for the 2^e candidates so that, by expectation, the guess of k_0 is correct in one of the attacks. The attacker has, however, no means of detecting whether the correct k_0 has been used and must continue until the output of the GGM tree to verify key guesses. The attack on the second 2-PRG will, hence, add the same amount of entropy, i.e. 2^e out of the full additional key entropy. As a result, after those two stages, a total entropy, or attack complexity, of $2^e \cdot 2^e = 2^{2e}$ is achieved. This can be generalized over n 2-PRG stages which results in a total remaining entropy of 2^{ne}. However, the entropy of the construction is upper bound by the length of k_{PRF}, i.e. 128 bit. The value of e, and, consequently, the number of required stages, is highly dependent on the exact implementation and can be estimated by conducting an attack on the final device.

As a note, the attack on the second 2-PRG and, optionally, subsequent 2-PRGs, differs in that the plaintexts are not carefully chosen but random. Hence, there is no correlated noise of S-boxes. But the simulation in Section 4 shows, that with data complexity 2, the expected guessing entropy per key byte is practically the same and we can disregard this difference.

Part 2: Security of the Unknown-Inputs GGM tree In a recent contribution, Unterluggauer et al. [16] describe, how the Unknown-Plaintext Template Attack [6], which is a second-order profiled DPA, can be modified to fit the case of leakage-resilient constructions with unknown inputs by switching the role of key and plaintext. Their goal was to retrieve unknown plain data from encryptions with frequent key updates. This directly applies to the unknown-inputs construction in [13]. The (constant) unknown plaintext is attacked and retrieved using templates on the unknown changing keys and the corresponding outputs of the S-box transformation. They present a practical attack on a microcontroller implementation of AES without parallel noise and succeed with about 2.000

traces. The changing keys are not recovered in this setting which is acceptable for their attack goal.

At first glance, this seems a potential threat also for our construction, specifically to the unknown-inputs GGM tree. However, their attack leads to the recovery of the unknown inputs only which cannot be directly used by an attacker to predict the PRFs output. Hence, a second first-order DPA attack using the resulting guesses for the plaintexts needs to be used to attack the key. This corresponds to an attack on the 2-PRG as discussed in the previous part 1. More importantly, contrary to the setting of Unterluggauer et al., the correlated algorithmic noise from the parallel setting is effective.

To address attacks on unknown inputs and key when such noise from parallelism is present, Medwed et al. [13] used simulations of second-order template attacks on the key using templates for the unknown plaintexts and the S-box outputs (see Fig. 5, right part in [13]). This experiment is equivalent to the attack described by Unterluggauer et al. only with switched roles for plaintexts and keys. The results of Medwed et al. [13] in Fig. 5, suggest that noise from 2 or 4 "overlapping" S-boxes is sufficient to achieve a guessing entropy per byte greater than 4, respectively 6. Considering our practical results, this is equivalent to at least 2, or 4 S-box signals overlapping at every location and point in time. This seems to be a reasonable requirement, as these effects are the same as the ones which are exploited in the first part and cause the remaining entropy of 2^e after an attack on the known-inputs 2-PRG. We therefore tend to believe that such attacks are unsuccessful in practice, but leave a thorough analysis for future work.

Finally, note that additional care has to be taken if the output of the PRF is used in an application where it is directly exposed to the attacker. Then an additional output whitening step at the end of the GGM tree is necessary where a fixed plaintext is encrypted. Otherwise the last step would be susceptible to an attack with two known ciphertexts, which is equivalent to the known input attack on the initial 2-PRG stage.

A Cautionary Note The security of the proposed construction is based on the fundamental assumption that enough entropy remains after localized EM attacks in the first 2-PRG step as shown by Unterstein et al. [17]. This assumption can only be verified empirically by proper laboratory side-channel evaluations. The number of stages can be configured according to the results of this analysis. If no entropy remains after localized EM attacks in the first stage, then our construction only increases the effort of the attacker who has to repeat measurements and attacks on the second and further stages.

7 Conclusion

In this work, we investigated the reasons *why* state-of-the-art localized EM attacks are able to successfully isolate the leakage of parallel S-boxes within LR-PRFs. The most important result in this respect is that not only the *high spatial*

resolution, but also the *high temporal resolution* is contributing to these isolation capabilities. This is somewhat a negative result for designers, as being able to fully control the timing characteristics of signals on FPGA devices seems to be unrealistic. As a positive result, we showed that the equal leakage assumption is not a necessary condition when the data complexity is limited to 2-inputs only, hence allowing for more compact LR-PRF implementations using interleaved placement and routing. Finally, we presented an extension to the unknown-inputs leakage-resilient PRF presented at ASIACRYPT 2016 which introduces additional key entropy to mitigate the entropy loss due to high-resolution EM attacks under verifiable empirical assumptions. It comes at a reasonable overhead and only requires additional key storage and no particularly stringent design constraints, i.e. it can be instantiated on devices with limited control over the underlying process technology, such as FPGAs.

We think our contribution is an important step towards securing implementations of leakage-resilient primitives on FPGAs in a practical and simple way. We encourage further investigation of LR-PRFs on ASIC devices in order to understand, how our results translate to other (and smaller) technologies which offer more controls on the timing characteristics of signals.

Acknowledgements. The work presented in this contribution was supported by the German Federal Ministry of Education and Research in the project *ALESSIO* through grant number 16KIS0629.

8 SNR for All S-boxes

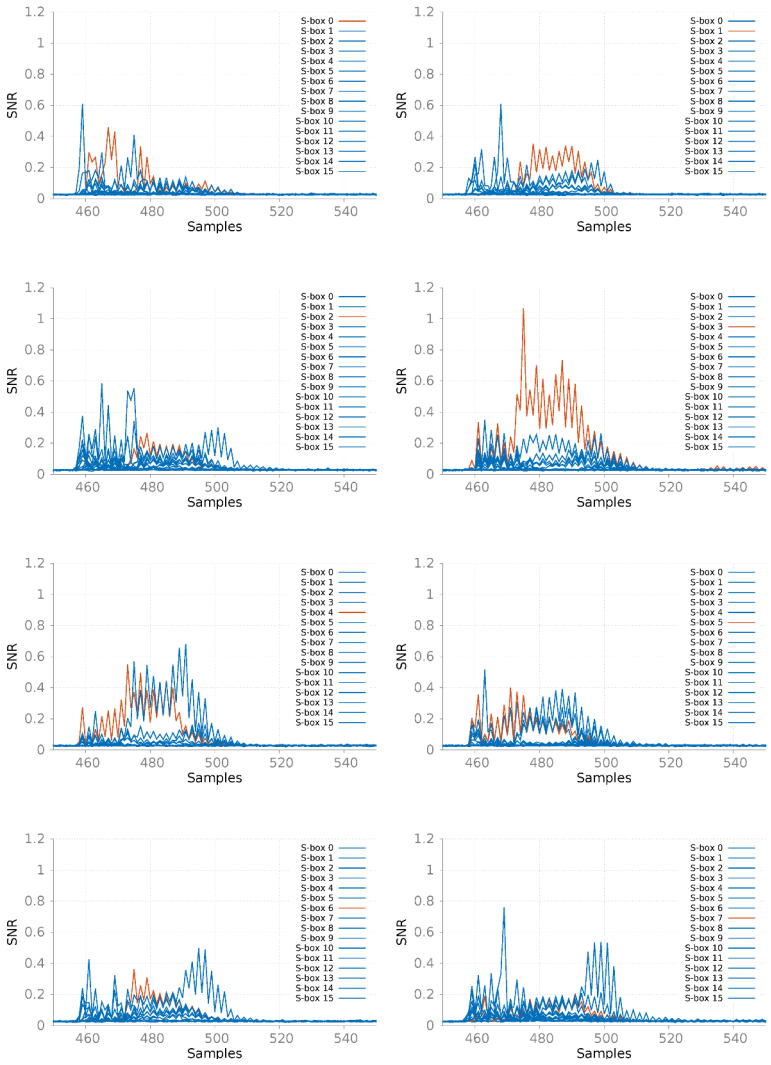


Fig. 9: SNR for S-boxes 0 to 7

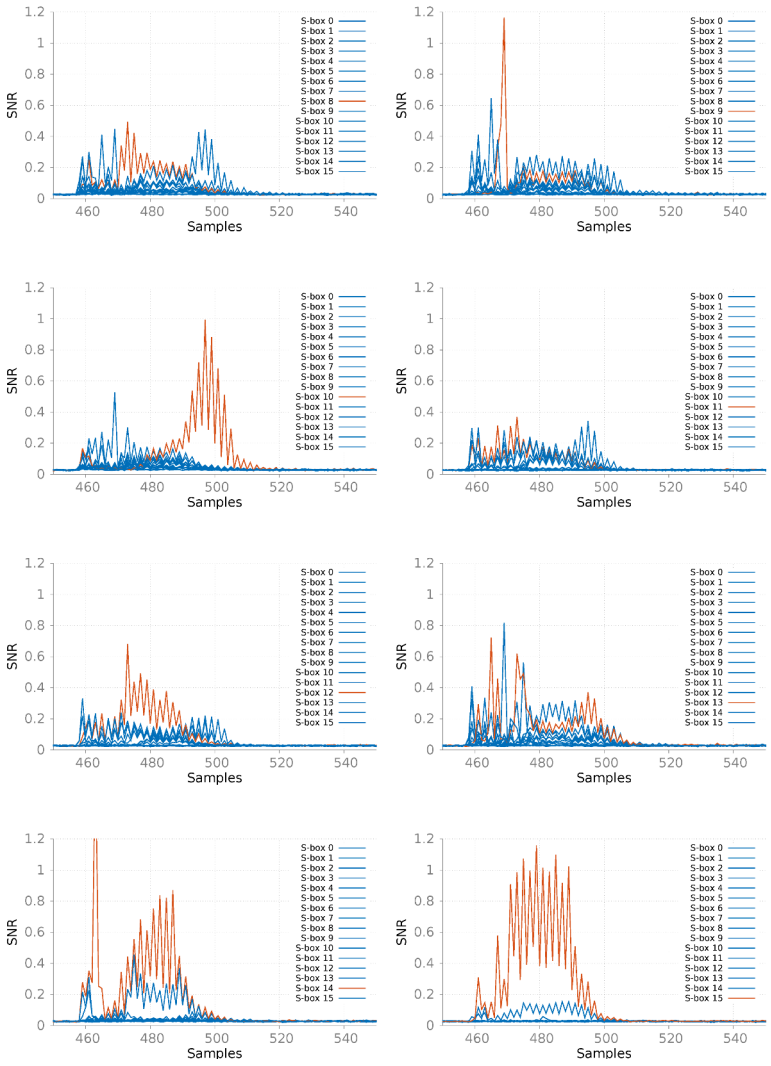


Fig. 10: SNR for S-boxes 8 to 15

References

1. Belaïd, S., De Santis, F., Heyszl, J., Mangard, S., Medwed, M., Schmidt, J.M., Standaert, F.X., Tillich, S.: Towards fresh re-keying with leakage-resilient PRFs: cipher design principles and analysis. Journal of Cryptographic Engineering 4(3), 157–171 (2014)
2. Bruneau, N., Guilley, S., Heuser, A., Marion, D., Rioul, O.: Less is more - dimensionality reduction from a theoretical perspective. In: Cryptographic Hardware

and Embedded Systems - CHES 2015 - 17th International Workshop, Saint-Malo, France, September 13-16, 2015, Proceedings. pp. 22–41 (2015)
3. Canright, D.: A very compact s-box for AES. In: Cryptographic Hardware and Embedded Systems - CHES 2005, 7th International Workshop, Edinburgh, UK, August 29 - September 1, 2005, Proceedings. pp. 441–455 (2005)
4. Chari, S., Jutla, C.S., Rao, J.R., Rohatgi, P.: Towards Sound Approaches to Counteract Power-Analysis Attacks, pp. 398–412. Springer Berlin Heidelberg, Berlin, Heidelberg (1999)
5. Goldreich, O., Goldwasser, S., Micali, S.: How to construct random functions. Journal of the ACM (JACM) 33(4), 792–807 (1986)
6. Hanley, N., Tunstall, M., Marnane, W.P.: Unknown plaintext template attacks. In: Information Security Applications, 10th International Workshop, WISA 2009, Busan, Korea, August 25-27, 2009, Revised Selected Papers. pp. 148–162 (2009)
7. Heyszl, J., Merli, D., Heinz, B., De Santis, F., Sigl, G.: Strengths and limitations of high-resolution electromagnetic field measurements for side-channel analysis. In: Mangard, S. (ed.) Smart Card Research and Advanced Applications. Lecture Notes in Computer Science, Springer Berlin Heidelberg (2012)
8. Immler, V., Specht, R., Unterstein, F.: Your rails cannot hide from localized EM: How dual-rail logic fails on FPGAs. In: International Conference on Cryptographic Hardware and Embedded Systems. pp. 403–424. Springer (2017)
9. Kirschbaum, M.: Power Analysis Resistant Logic Styles – Design, Implementation, and Evaluation. Ph.D. thesis (2011)
10. Mangard, S., Oswald, E., Popp, T.: Power Analysis Attacks. Springer Science & Business Media (2008)
11. May, D., Muller, H.L., Smart, N.P.: Non-deterministic Processors, pp. 115–129. Springer Berlin Heidelberg, Berlin, Heidelberg (2001)
12. Medwed, M., Standaert, F., Joux, A.: Towards super-exponential side-channel security with efficient leakage-resilient PRFs. In: Prouff, E., Schaumont, P. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2012 - 14th International Workshop, Leuven, Belgium, September 9-12, 2012. Proceedings. Lecture Notes in Computer Science, vol. 7428, pp. 193–212. Springer (2012)
13. Medwed, M., Standaert, F.X., Nikov, V., Feldhofer, M.: Unknown-input attacks in the parallel setting: Improving the security of the CHES 2012 leakage-resilient PRF. In: Advances in Cryptology–ASIACRYPT 2016: 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I. pp. 602–623. Springer (2016)
14. Standaert, F.X., Pereira, O., Yu, Y., Quisquater, J.J., Yung, M., Oswald, E.: Leakage resilient cryptography in practice. IACR Cryptology ePrint Archive 2009, 341 (2009)
15. Standaert, F.X., Pereira, O., Yu, Y.: Leakage-resilient symmetric cryptography under empirically verifiable assumptions. In: Advances in Cryptology–CRYPTO 2013, pp. 335–352. Springer Berlin Heidelberg (2013)
16. Unterluggauer, T., Werner, M., Mangard, S.: Side-channel plaintext-recovery attacks on leakage-resilient encryption. In: Design, Automation Test in Europe Conference Exhibition (DATE), 2017. pp. 1318–1323 (March 2017)
17. Unterstein, F., Heyszl, J., De Santis, F., Specht, R.: Dissecting leakage resilient PRFs with multivariate localized EM attacks - a practical security evaluation on FPGA. In: Proceedings of 8th International Workshop on Constructive Side-Channel Analysis and Secure Design (COSADE 2017). Springer (2017)