Lightweight Ciphers and their Side-channel Resilience

Heuser, Annelie ; Picek, Stjepan; Guilley, Sylvain; Mentens, Nele

DOI
10.1109/TC.2017.2757921

Publication date
2020

Document Version
Accepted author manuscript

Published in
IEEE Transactions on Computers

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

Lightweight Ciphers and their Side-channel Resilience

Annelie Heuser, Stjepan Picek, Sylvain Guilley, and Nele Mentens

Abstract—Side-channel attacks represent a powerful category of attacks against cryptographic devices. Still, side-channel analysis for lightweight ciphers is much less investigated than for instance for AES. Although intuition may lead to the conclusion that lightweight ciphers are weaker in terms of side-channel resistance, that remains to be confirmed and quantified. In this paper, we consider various side-channel analysis metrics which should provide an insight on the resistance of lightweight ciphers against side-channel attacks. In particular, for the non-profiled scenario we use the theoretical confusion coefficient and empirical optimal distinguisher. Our study considers side-channel attacks on the first, the last, or both rounds simultaneously. Furthermore, we conduct a profiled side-channel analysis using various machine learning attacks to recover 4-bit and 8-bit intermediate states of the cipher. Our results show that the difference between AES and lightweight ciphers is smaller than one would expect, and even find scenarios in which lightweight ciphers may be more resistant. Interestingly, we observe that the studied 4-bit S-boxes have a different side-channel resilience, while the difference in the 8-bit ones is only theoretically present.

Index Terms—Side-channel analysis, lightweight ciphers, optimal distinguisher, confusion coefficient, success rate, machine learning attacks.

1 Introduction

With the advent of the Internet of Things, we are surrounded with smart objects (aka things) that have the ability to communicate with each other and with centralized resources. The two most common and widely noticed artifacts are RFID and Wireless Sensor Networks which are used in supply-chain management, logistics, home automation, surveillance, traffic control, medical monitoring, and many more applications. Most of these applications have the need for cryptographic secure components which inspired research on cryptographic algorithms for constrained devices. Accordingly, lightweight cryptography has been an active research area over the last 10 years. A number of innovative ciphers have been proposed in order to optimize various performance criteria and have been subject to many comparisons.

Furthermore, lightweight cryptography is also an enabler in forthcoming technologies, such as 5G communications and connected cars, which are expected to be deployed in 2020 (e.g., corresponding, in Japan, to Tokyo Olympic and Paralympic Games). Indeed, 5G specifications target end-to-end security with < 1 ms latency, hence ultra-low delay cryptographic primitives are needed. Besides, connected cars shall interact with the infrastructure in both an authenticated and timely manner: actually, an accident can occur if a driving decision in delayed by more than a few milliseconds. Therefore, lightweight cryptography is expected to find industrial applications in the near future. This is why secured (say, validated according to either Common Criteria or FIPS 140) lightweight cryptography is a topic of interest, which we address in depth in this paper.

In particular, the resistance against side-channel attacks has been considered as an additional decision factor lately. Side-channel attacks analyze physical leakage that is unintentionally emitted during cryptographic operations in a device (e.g., power consumption [1], electromagnetic emanation [2]). This side-channel leakage is statistically dependent on intermediate processed values involving the secret key, which makes it possible to retrieve the secret from the measured data. So-called profiled side-channel distinguishers assume that the attacker is able to possess an additional device to the one he wants to attack, and on which he has the freedom of nearly full control. In this advanced setting, Machine learning (ML) techniques have shown to be effective in various scenarios (e.g., [3], [4]).

Side-channel analysis for lightweight ciphers is of particular interest not only because of the apparent lack of research so far, but also because of the interesting properties of S-boxes. Since the nonlinearity property for S-boxes usually used in lightweight ciphers (i.e., 4×4) can be maximally equal to 4, the difference between the input and the output of an S-box is much smaller than for instance for AES [5]. Therefore, one could conclude that from that aspect, SCA for lightweight ciphers must be more difficult. However, the number of possible classes (e.g., Hamming weight (HW) or key classes) is significantly lower, which may indicate that (profiled) SCA must be easier than for standard ciphers. Besides the difference in the number of classes and consequently probabilities of correct classification, there is also a huge time and space complexity advantage (for the attacker) when dealing with lightweight ciphers.

1.1 Our Contributions

In this paper we give a detailed study of lightweight ciphers in terms of side-channel resistance, in particular for software implementations. As a point of exploitation we concentrate on the non-linear operation (S-box) during the first round, the last round, and both round simultanously (which is particular interest the cipher uses the same key in the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

first and last round and the S-box is not involutive[1]). Our comparison includes SPN ciphers with 4-bit S-boxes such as KLEIN [6], Midori [7], Mysterion [8], LED [9], Piccolo [10], PRESENT [11], PRIDE [12], PRINCE [13], RECTANGLE [14], Skinny [15] as well as ciphers with 8-bit S-boxes: AES, Zorro [16], Robin [17].

In the non-profiled scenario we investigate first the relationship between different key hypotheses with the confusion coefficient [18], [19]. Using specific properties of the confusion coefficient (like the minimum value and the variance) we give a preliminary classification regarding the side-channel resistance. Furthermore, using simulated data for various signal-to-noise ratios (SNR) we present empirical results for the optimal distinguisher [20] and discuss the difference between attacking 4-bit and 8-bit S-boxes. Finally, we compare several supervised (i.e., profiled) machine learning techniques in order to recover 4-bit and 8-bit intermediate states. These results are of particular interest when conducting algebraic side-channel analysis [21]. This paper is an extended version of a paper published in [22].

1.2 Road Map

This paper is organized as follows. Section 2 gives basic information on the ciphers and exploitations we investigate. Next, in Section 3 we discuss the optimal distinguisher, confusion coefficient, and conduct empirical evaluations to reveal the secret (round)key. In Section 4 we use profiled machine learning side-channel analysis to recover intermediate states. Section 5 concludes and offers directions for future work.

2 Ciphers & Exploitations

2.1 Investigated Ciphers

2.1.1 AES [5]

The Advanced Encryption Standard (AES) has been standardized by NIST in 2001 [23]. It has an SPN structure with an internal fixed block size of 128-bits represented as a 4×4 byte matrix. At the beginning, the plaintext state is `xor-ed` with the secret key. Subsequently, each encryption round consists of the application of `SubBytes`, `ShiftRows`, `MixColumns`, and `AddRoundKey`, in the last round, `MixColumns` is omitted.

2.1.2 KLEIN [6]

KLEIN is an AES-like lightweight block cipher. The substitution stage uses 16 similar involutive 4-bit S-boxes. Similar to AES, each encryption round consists of `AddRoundKey`, `SubNibbles`, `RotNibbles`, and `MixNibbles`, followed by a final key addition.

2.1.3 LED [9]

LED is heavily based on AES. The encryption is divided in steps which consists in 4 rounds and a xor operation with the key. Each round is made of the xoring of a round constant and AES-style `SubCells`, `ShiftRows` and `MixColumnsSerial` operations. The `S-box` used in the `SubCells` step is the PRESENT `S-box`. Interestingly, LED does not have a key scheduling: a key of 64 bits is xored with internal state. For the 128-bit version the key is divided into two subkeys of 64 bits which are used alternatively.

2.1.4 Mysterion [8]

The Mysterion cipher is one instance of the so-called LS-design, in which the internal state of the cipher is a matrix of $s \times L$ bits. The internal state of the block cipher is organized into a 4×32 bit matrix for Mysterion-128, which is further subdivided into 4 4×8 blocks. A round contains the following operations: `S-box layer`, `L-Box layer` and `ShiftColumns`. The `S-box layer` is a 4-bit S-box called "Class 13", as introduced in [24], that is applied in parallel to each column of the internal state.

2.1.5 Piccolo [10]

Piccolo is a Generalized Feistel Network with 4 16-bit branches using an advanced permutation (diffusion layer) as well as whitening. The 4-bit `S-box` has a decent non-linearity and differential uniformity, while having a tiny hardware footprint: it can be implemented using only 4 NOR gates, 3 XOR gates and 1 XNOR gate.

2.1.6 PRESENT [11]

PRESENT has a 64-bit block size with a bit oriented permutation layer. The non-linear layer is based on a single 4-bit S-box which was designed to be optimal in hardware. An encryption round consists of `AddRoundKey`, a substitution (`sBoxLayer`), and a permutation layer (`pLayer`). A final key addition is performed after the encryption rounds.

2.1.7 PRIDE [12]

PRIDE has been optimized for 8-bit microcontrollers with a special focus on the linear layer of the cipher. It is designed in a bit-sliced fashion to minimize the number of instructions necessary to evaluate it. The 4-bit `S-box` is an involution.

2.1.8 PRINCE [13]

The main aim of the design of PRINCE is to provide low latency. It has a small number of rounds and the layers in a round have low logic depth. The cipher uses no real key schedule. The core function contains 5 "forward" rounds, a middle round and then 5 "backward" rounds, so 11 rounds in total. A forward round starts by a `xor` with a round constant xor key, then a non linear layer `S` and then a linear layer `M`. The "backward" rounds are exactly the inverse of the "forward" rounds except for the round constants.

2.1.9 RECTANGLE [14]

The state of RECTANGLE is represented as a 4×16 matrix. The non-linear layer consists of the parallel application of a 4-bit S-box on the columns of the state and the linear layer is a fixed rotation over a different amount of steps in each row.

1. The S-box is not equal to its inverse.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

2.1.10 Robin [17]

Like Mysterion, Robin is based on the LS-design principles. The non-linear layer consists of a parallel applications of a $s \times s$ bits $(s = 8)$ permutation on each column, which is chosen to be efficiently implemented in a bit-sliced fashion and an involution. The linear layer consists of the application of a linear $L \times L$ bit $(L = 16)$ permutation on each row of the matrix.

2.1.11 Skinny [15]

SKINNY is a family of adaptable lightweight block ciphers designed such that the hardware footprint is small. All members of the family are SPN consisting of several iterations of the following operations transforming a 4x4 matrix of 4-bit nibbles (64-bit variant) (considered in this paper) or bytes (128-bit variant). It consists of the following operations: `SubCell` (non-linear), `AddConstants`, `ShiftRows`, and `MicColumns`.

2.1.12 Zorro [16]

Zorro is a modified version of AES with a variant of the S-box that is easier to mask. Fewer `S-box` calls are performed and the number of multiplications has been minimized. Besides, the execution is split into "steps" of 4 rounds and the key (simply the master key) is added only at the end of each step.

2.2 Exploitations

In this paper, our main targets are the weaknesses arising in software implementations on serial microprocessors. In these applications, the Hamming weight (HW) and the Hamming distance (HD) leakage model are most commonly found in practice. More precisely, the loading and storing of data in memory (e.g., `S-box` calls) is usually causing HW leakage, whereas the register updating (e.g., writing of intermediate round states) is causing HD leakage. Typically the latter is less significant than the former, which is why we concentrate on a specific memory operation. Moreover, a classical point of exploitation for side-channel analysis is the first or last round, as in these outer rounds the amount of key hypothesis to be made is rather small and thus efficiently enumerable.

Note that our study does not include leakages from all kinds of operations in the specific ciphers, nor (in case the cipher uses a key scheduling algorithm) the complexity to go from a round key to the master key, which may be an interesting next step for future work.

2.2.1 First Round

The main common operation all previous described ciphers share, is first the addition (`xor`) of the roundkey/masterkey followed by (at least one) S-box call. When concentrating on the first round our study therefore concentrates on leakage measurements X arising from an S-box lookup operation as

$$X = \alpha \cdot \mathrm{HW}(\mathtt{Sbox}[P \oplus k^\star]) + N, \tag{1}$$

where N is independent additive Gaussian noise with variance σ^2, $k^\star$ one chunk of the secret key (first round key or master key), P a plaintext chunk (byte or nibble), and α is a scaling factor.

2.2.2 Last Round

When attacking the last round the attacker uses the ciphertext to make hypotheses about the state of the S-box input, i.e. leakage arising as

$$X = \alpha \cdot \mathrm{HW}(\mathtt{Sbox}^{-1}[C \oplus k^\star]) + N, \tag{2}$$

where $k^\star$ one chunk of the secret key (last round key or master key).

3 Recovering the (Round) Key

In this section we are interested in recovering a chunk of the key used in the first or in the last round. For this we first consider the leakage of the first and last round independently and then also in combination. To determine the worst-case scenario (most powerful attacker) we use the optimal distinguisher [20] and highlight which properties are influencing its success exponent [25] which is the first-order exponent of the success rate. Our results are confirmed by empirical evaluations of the success rates.

3.1 Optimal Distinguisher & Theoretical Success rate

The optimal distinguisher in case the leakage is known in a direct scale[2] and the noise is Gaussian is defined as

$$D(k) = -(X - \alpha Y(k))^2, \tag{3}$$

where $Y(k)$ is the predicted intermediate state depending on a key guess k. More precisely, when considering the S-box output in the first round (see Eq. (1))

$$Y(k) = \mathrm{HW}(\mathtt{Sbox}[P \oplus k]), \tag{4}$$

whereas when attacking the last round (see Eq. (2)) we have

$$Y(k) = \mathrm{HW}(\mathtt{Sbox}^{-1}[C \oplus k^\star]). \tag{5}$$

From Eq. (3) using the maximum likelihood rule an attacker predicts the secret key guess

$$\hat{k} = \arg\max_k D(k). \tag{6}$$

The most common measure for side-channel evaluation is the empirical success rate SR which is the probability of success given a certain amount of leakage measurements. Interestingly, the authors in [25] showed that for any side-channel attack the SR can be modeled using a *first-order exponent* (SE) [26], i.e. there exists a constant SE such that[3]

$$1 - \mathsf{SR} \approx \exp(-q \cdot \mathsf{SE}),$$

where q is the number of traces for the expected success rate to be equal to SR.

Now, the first-order exponent SE for the optimal distinguisher takes the following form [25]

$$\mathsf{SE} = \min_{k \neq k^\star} \frac{1}{2} \frac{\kappa(k^\star, k)^2}{\kappa''(k^\star, k) - \kappa(k^\star, k)^2 + \kappa(k^\star, k)/\mathrm{SNR}}, \tag{7}$$

2. The scaling factor α is known or well enough approximated.

3. We use the same definition as in [25]: a function $f(x)$ has *first order exponent* $\xi(x)$ if $\big(\ln f(x)\big)/\xi(x) \to 1$ as $x \to +\infty$, in which case we write $f(x) \approx \exp \xi(x)$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

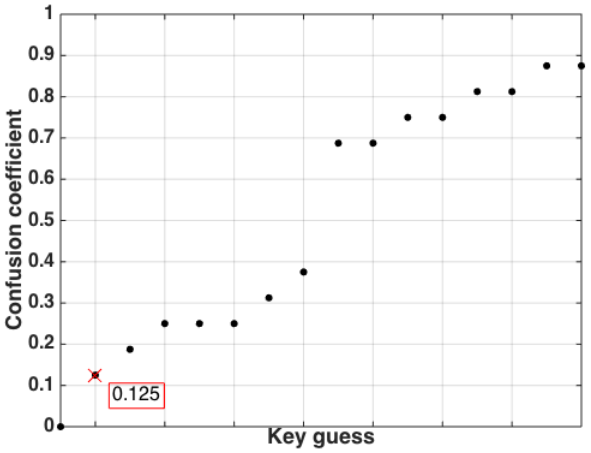

Fig. 1: KLEIN

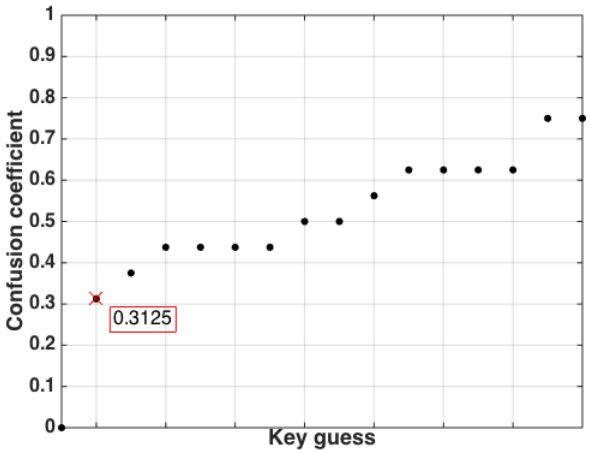

Fig. 2: Mysterion

where SNR $= \frac{\alpha^2}{\sigma^2}$ is the signal-to-noise-ratio and

$$\kappa(k^\star, k) = \mathbb{E}\Big\{\Big(\frac{Y(k^*) - Y(k)}{2}\Big)^2\Big\}, \tag{8}$$

$$\kappa''(k^\star, k) = \mathbb{E}\Big\{\Big(\frac{Y(k^*) - Y(k)}{2}\Big)^4\Big\} \tag{9}$$

are two versions of confusion coefficients (which generalize that of [18]). Loosely speaking, the confusion coefficients measure the dependencies between the prediction of the intermediate states of the secret key $k^\star$ with any key hypothesis k.

When the SNR is low, then Eq. (7) simplifies to

$$\mathsf{SE} \approx \frac{1}{2} \min_{k \neq k^\star} \kappa(k^\star, k) \cdot \mathrm{SNR}. \tag{10}$$

Accordingly, considering the described leakages in Eq. (1) and Eq. (2) one can see from Eq. (8) and (9) that the confusion coefficient depends on the particular choice of the S-box and therefore does SE and SR.

Next, we will focus on the confusion coefficient $\kappa(k^\star, k)$ and give empirical results for the success rate for all the previously described ciphers.

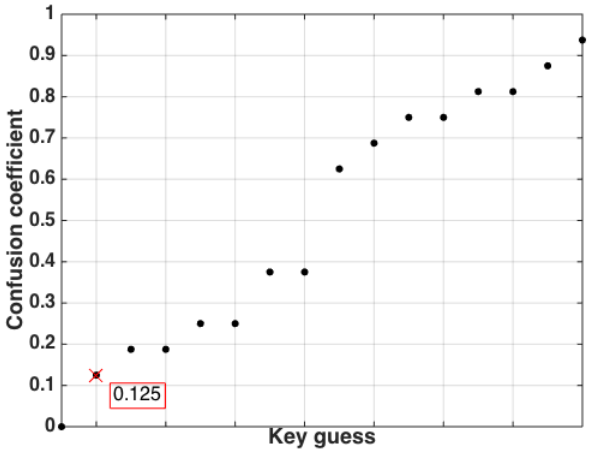

Fig. 3: Midori 1

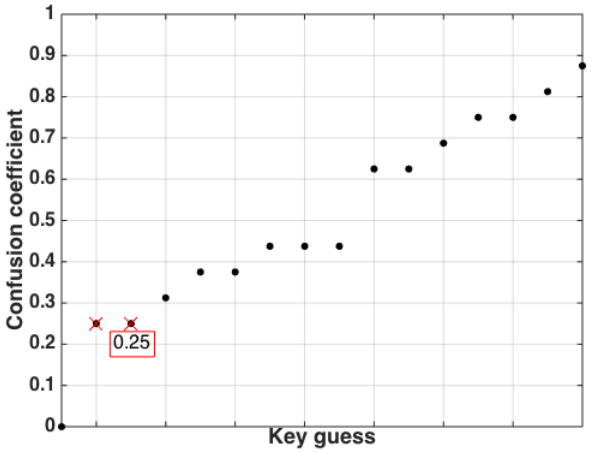

Fig. 4: Midori 2

3.2 Attacking the first Round

3.2.1 *Confusion Coefficients*

Figures 1 to 9 show the confusion coefficient for 4-bit S-boxes and Figures 11 to 13 for 8-bit S-boxes. Note that the distribution of $\kappa(k^\star, k)$ is independent on the particular choice of $k^\star$ (in the case there are no weak keys) and the values are only permuted. For our experiments we choose $k^\star = 0$ and furthermore order $\kappa(k^\star, k)$ in an increasing order of magnitude. One can observe that the distribution is indeed different for the investigated ciphers. Note that if $\kappa(k^\star, k_1) = \kappa(k^\star, k_2)$ the optimal distinguisher is not able to distinguish between the key hypothesis k_1 and k_2.

We highlight $\min_{k \neq k^\star} \kappa(k^\star, k)$ with a red cross and state its value next to it. Recall from Eq. (10) that the minimum confusion coefficient is the influencing factor related to the S-box influencing the SE and SR (in case of reasonably low SNR). Comparing the minimum value for 4-bit S-boxes we

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

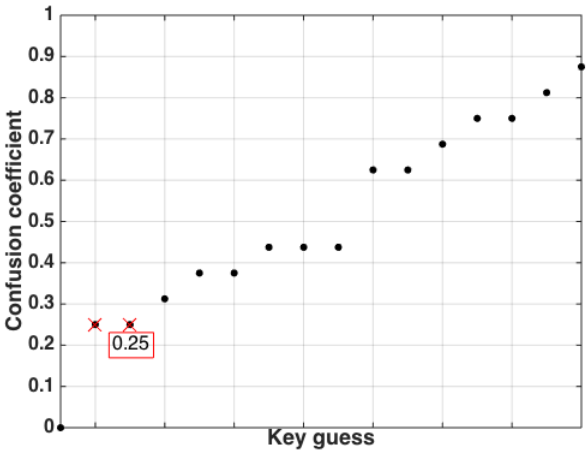

Fig. 5: PRESENT / LED

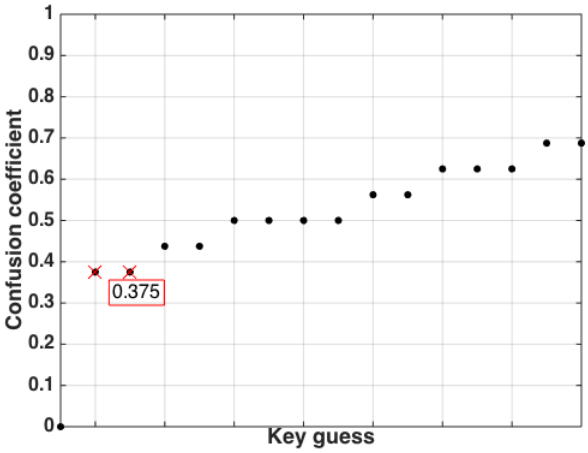

Fig. 6: Piccolo

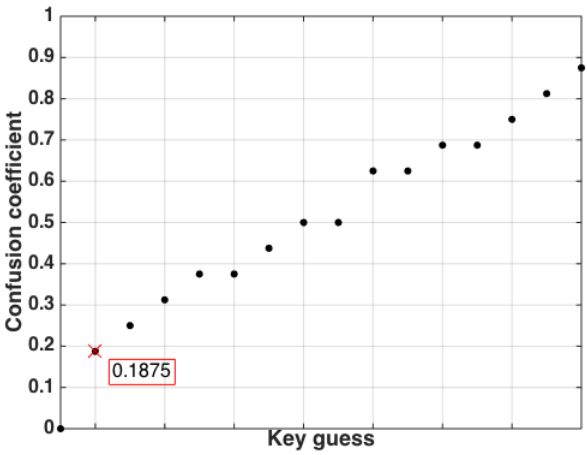

Fig. 7: PRINCE

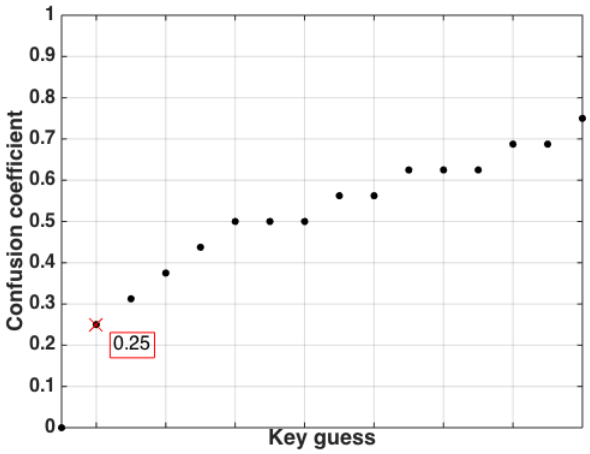

Fig. 8: PRIDE

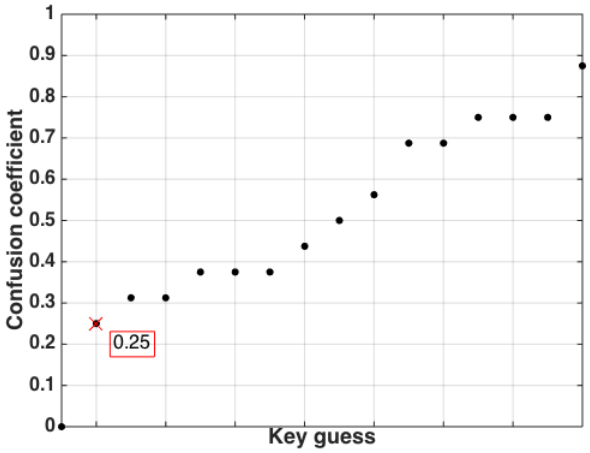

Fig. 9: RECTANGLE

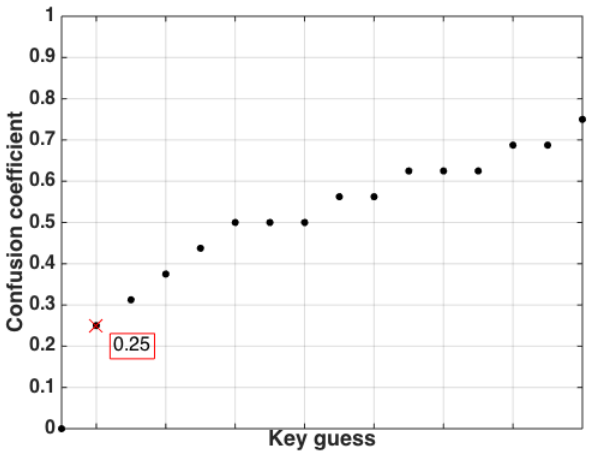

Fig. 10: Skinny

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

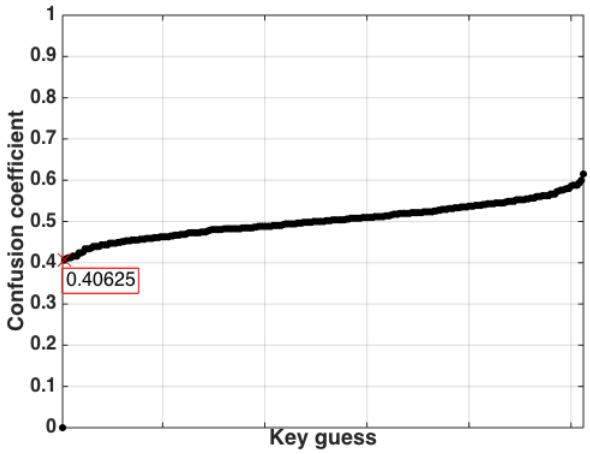

Fig. 11: AES

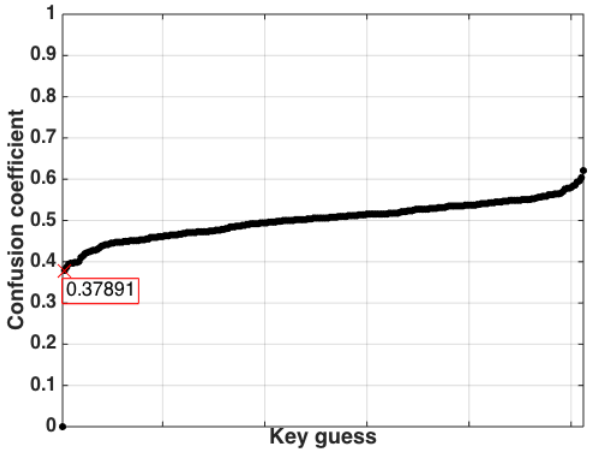

Fig. 12: Zorro

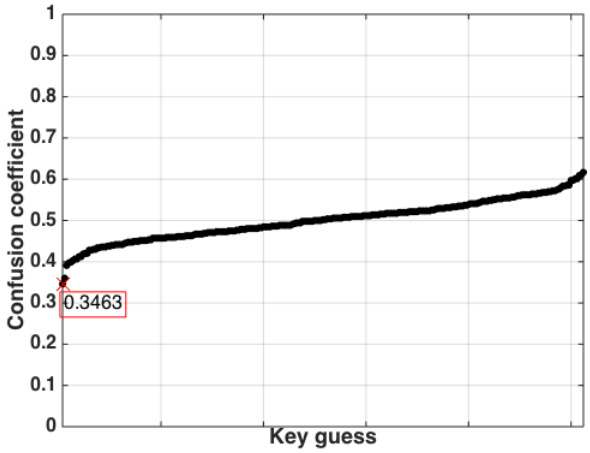

Fig. 13: Robin

achieve the following ranking (from weak to more side-channel resistant):

1) Piccolo,
2) Mysterion,
3) PRESENT/LED, PRIDE, RECTANGLE, Skinny, Midori 2,
4) PRINCE,
5) KLEIN, Midori 1.

Interestingly, the values given for 8-bit S-boxes indicate that the side-channel resistance of the investigated 8-bit S-boxes is lower than for the ones with 4-bit S-boxes. Recall that the confusion coefficient measures the relationship between different key hypotheses. Now, as for 8-bits we have 256 possible values for $T \in \mathbb{F}_2^8$ and $Y(k) \in [0, 1, \ldots, 8]$ it is easier to distinguish than for 4-bit S-boxes with $T \in \mathbb{F}_2^4$ and $Y(k) \in [0, 1, \ldots, 4]$.

However, in practice we cannot straightforwardly conclude that due to the properties of the confusion coefficient, 4-bit S-boxes are harder to attack than 8-bit S-boxes. One reason is that the confusion coefficient is theoretical (i.e., holding for $Q \to \infty$). But, especially for low noise scenarios Q might be small (below 100). So, naturally the 4-bit variant with only 16 inputs should converge faster than with 256 inputs. Or in other words, considering $Q = 100$, one can observe each plaintext for 4-bit S-boxes approximately 6.25 times, whereas for the 8-bit case more than the half has not been observed yet. Another reason is that the variance of the signal is not equivalent and thus the SNR in Eq. (10). In particular, as the HW follows a binomial distribution, we have $Var(\mathrm{HW}(\texttt{Sbox}[T \oplus k]))$ with $T, k \in \mathbb{F}_2^4$ equal to 1 for 4-bit S-boxes and equal to 2 for 8-bit S-boxes. Accordingly, given the same amount of independent additional noise, the SNR using 8-bit S-boxes is twice as high as for 4-bit S-boxes.

3.2.2 *Empirical Success Rate*

Figures 14 to 21 give the success rate for the optimal distinguisher for various levels of noise, where we simulated the traces as in Eq. (1) with $N \sim \mathcal{N}(0, \sigma^2)$ and $\alpha = 1$. To be reliable, we use 5 000 independent experiments with randomly chosen T. For 4-bit S-boxes, Figure 14 to Figure 17 confirms the ranking given by the confusion coefficient and listed above (Piccolo is the weakest and KLEIN, Midori 1 are the most resistant). It hold particularly for higher noise, which is inline with the theoretical derivations in Subsect. 3.1.

Figures 18 to 21 show that all three ciphers with 8-bit S-boxes behave similarly even for different levels of noise. Accordingly, the (small) differences in the minimum confusion coefficient do not influence the side-channel resistance in practice.

There are two ways to compare the success rates for 4-bit and 8-bit S-boxes, either having the same additional independent noise (environmental noise) σ or the same SNR. Using the same amount of σ (Figures 15 vs. 18 and 17 vs. 20), we can observe that AES, Zorro, and Robin are weaker than KLEIN/ Midori 1 and similar to or slightly worse than the others. On the other hand, when comparing the SNR, we observe that AES, Zorro, and Robin behave in a similar way as KLEIN/ Midori 1.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

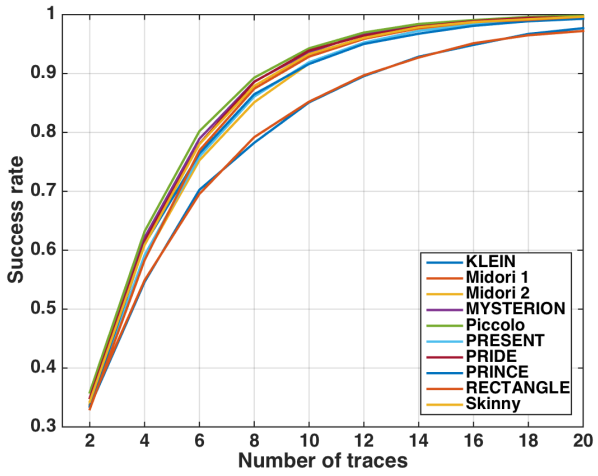

Fig. 14: $\sigma = \sqrt{1/2}$, SNR = 2 (first round)

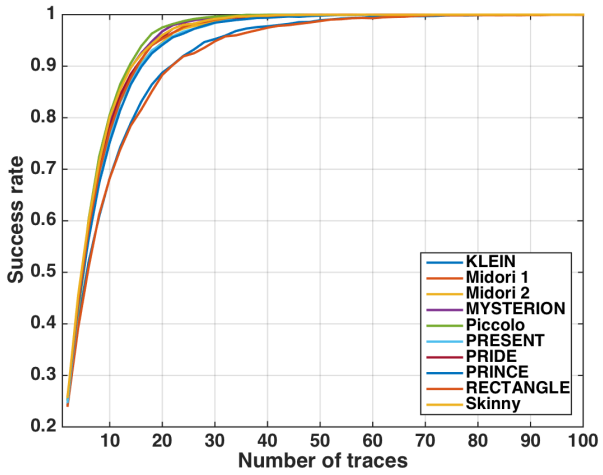

Fig. 15: $\sigma = 1$, SNR = 1 (first round)

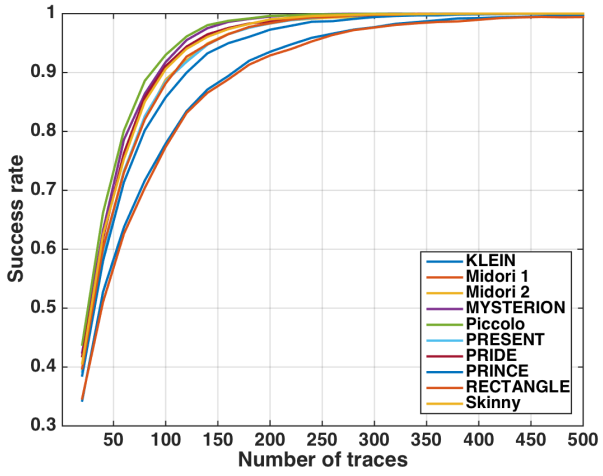

Fig. 16: $\sigma = \sqrt{8}$, SNR = $1/8$ (first round)

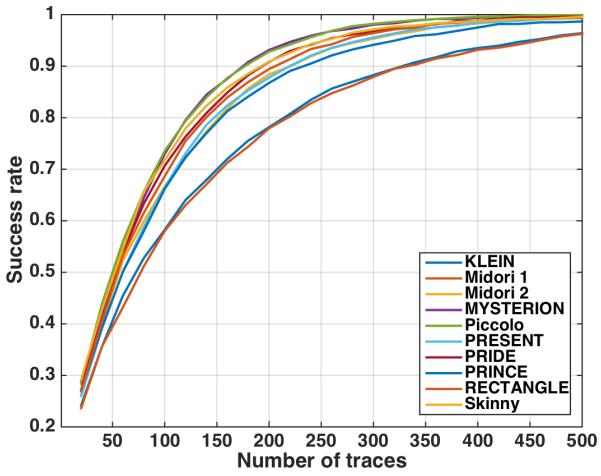

Fig. 17: $\sigma = 4$, SNR = $1/16$ (first round)

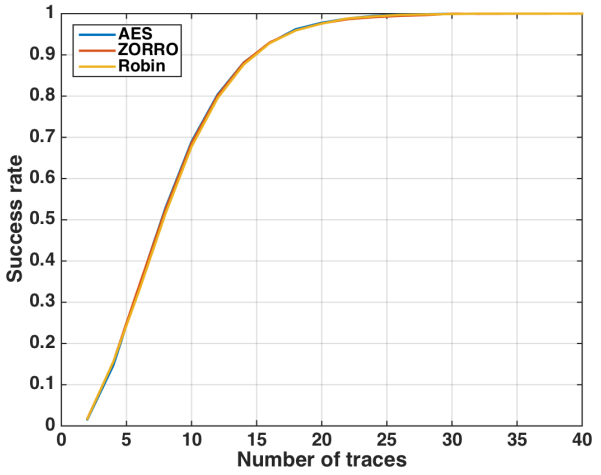

Fig. 18: $\sigma = 1$, SNR = 2 (first round)

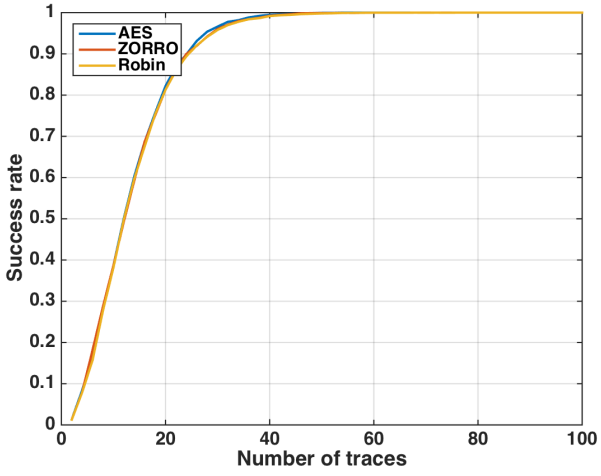

Fig. 19: $\sigma = \sqrt{2}$, SNR = 1 (first round)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

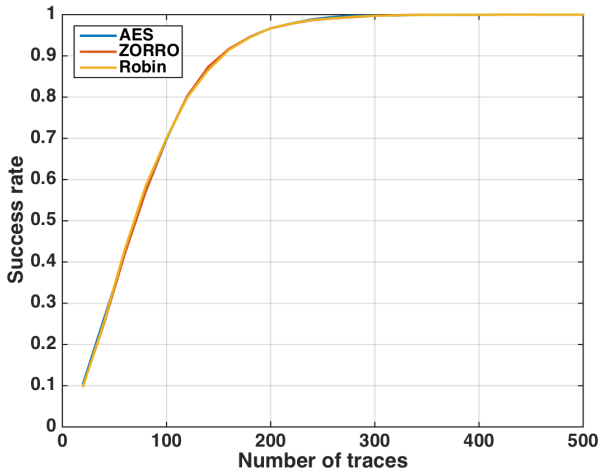

Fig. 20: $\sigma = 4$, SNR = $1/8$ (first round)

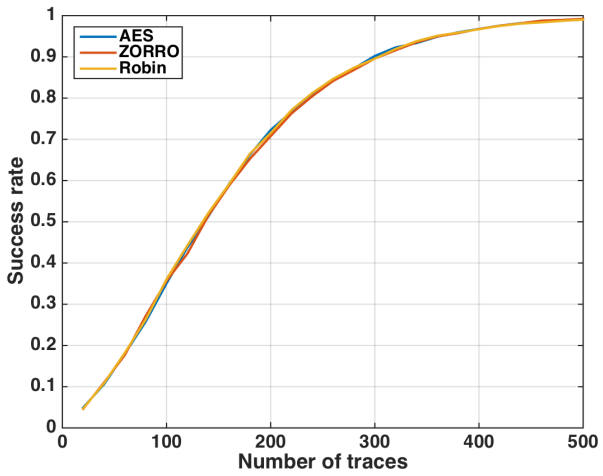

Fig. 21: $\sigma = \sqrt{32}$, SNR = $1/16$ (first round)

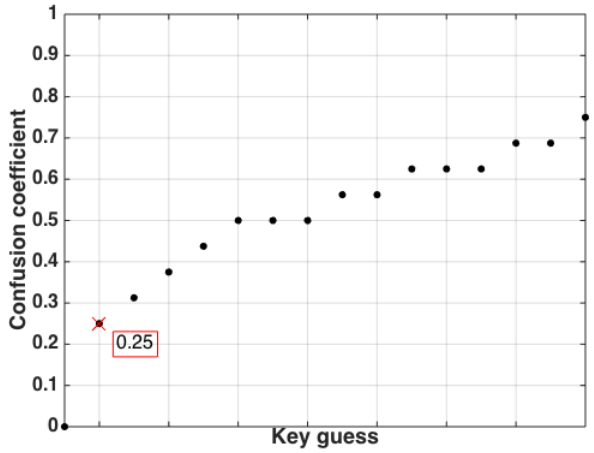

Fig. 22: Piccolo (inverse)

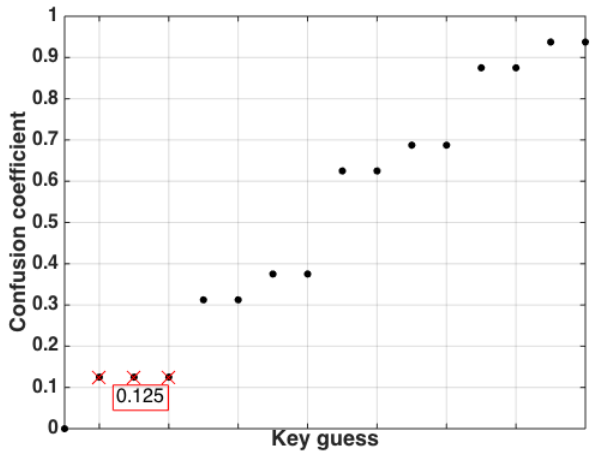

Fig. 23: PRESENT / LED (inverse)

3.3 Attacking the Last Round (inverse S-box)

Instead of attacking the first round using the plaintext, an attacker may also choose to attack the last round using the ciphertext. Accordingly, as he makes predictions about the S-box input in the last round (see Eq. (2)) we are now interested in properties of the inverse S-box. Note that KLEIN, Pride, Midori 1, Midori 2, Robin are involutions, which means that their S-box equals its inverse. For all remaining Sboxes we will first plot the confusion coefficients using the same methodology as before and then compare their empirical success rates.

3.3.1 Confusion Coefficient

Figures 22 to 28 illustrate the confusion coefficients for all non-involutive S-boxes. Interestingly, one can observe that compared to the results attacking the first round the minimum value of the confusion coefficient for Mysterion and Prince does not change, however, their distribution does. For all other investigated 4-bit S-boxes the inverse has a lower minimum confusion coefficient than for the direct S-box. For the 8-bit S-boxes AES has a lower minimum confusion coefficient whereas Zorro has a higher one when considering the inverse S-box. Accordingly, except for Zorro, we expect that when attacking the last round the empirical success rate should be less or equal to the success rate of the first round. For comparison all minimum confusion coefficients for the direct S-box and the inverse are additionally listed in Table 1. For 4-bit S-boxes (inverse) we achieve the following ranking (from weak to more side-channel resistant):

1) Mysterion
2) Midori 2, Piccolo, PRIDE, SKINNY
3) PRINCE,
4) KLEIN, Midori 1, PRESENT, RECTANGLE.

3.3.2 Empirical Success Rate

Figures 29 to 36 show the empirical success rate using the same simulation settings as previously. Interestingly, the success rates of the various ciphers are more distinctive than attacking the first round. Again we see a similar ranking as indicated by the minimum confusion coefficient and,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

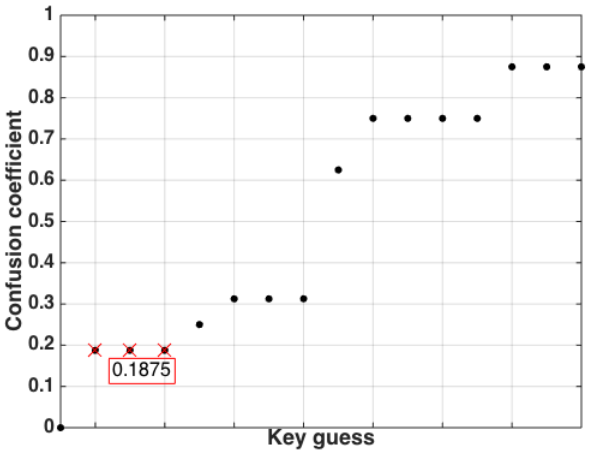

Fig. 24: Prince (inverse)

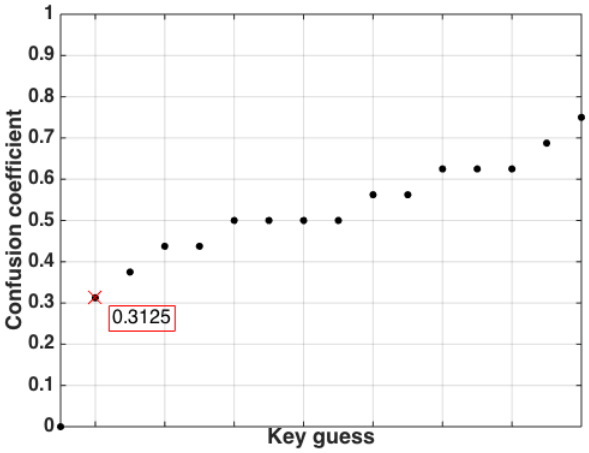

Fig. 25: MYSTERION (inverse)

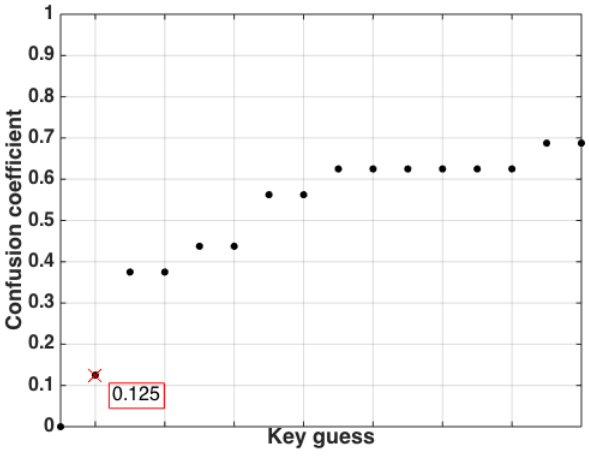

Fig. 26: Rectangle (inverse)

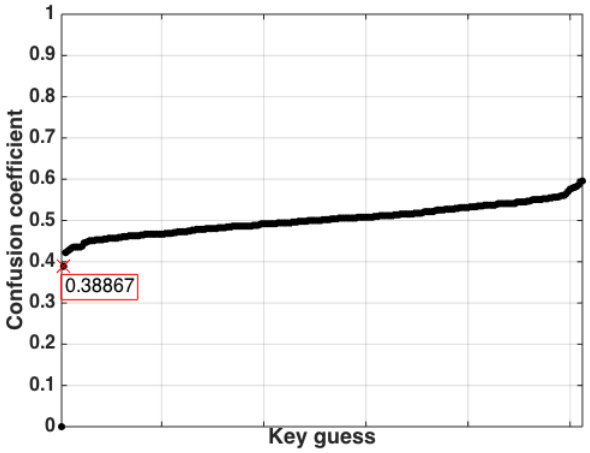

Fig. 27: AES (inverse)

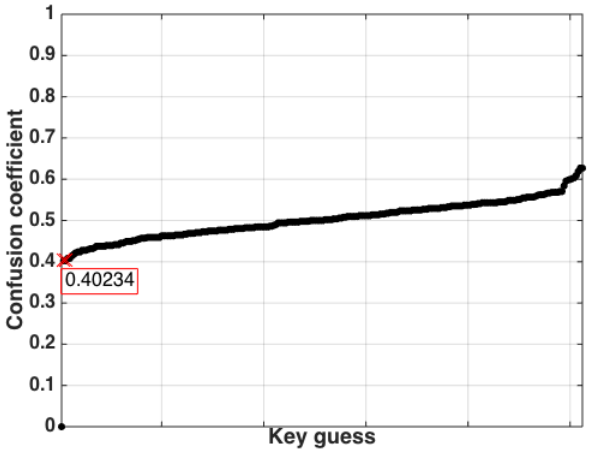

Fig. 28: Zorro (inverse)

TABLE 1: Minimum confusion coefficient

Name	involution	S-box	inverse S-box
KLEIN	x	0.125	0.125
Midori 1	x	0.125	0.125
Midori 2	x	0.250	0.250
Mysterion		0.3125	0.3125
Piccolo		0.375	0.25
PRESENT/LED		0.25	0.125
PRIDE	x	0.25	0.25
PRINCE		0.1875	0.1875
RECTANGLE		0.250	0.125
SKINNY	x	0.250	0.250
AES		0.406	0.388
Robin	x	0.347	0.347
Zorro		0.378	0.402

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

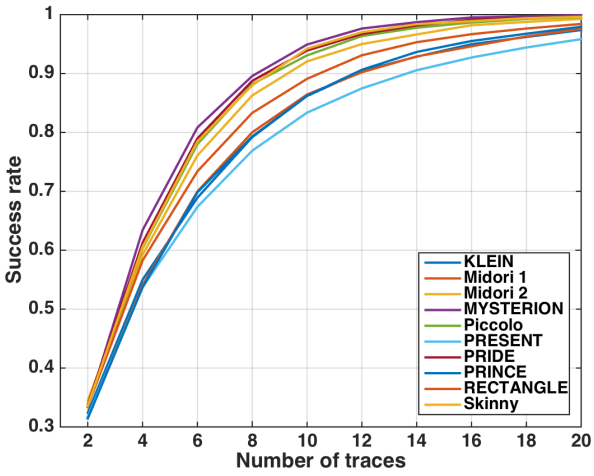

Fig. 29: $\sigma = \sqrt{1/2}$, SNR = 2 (last round)

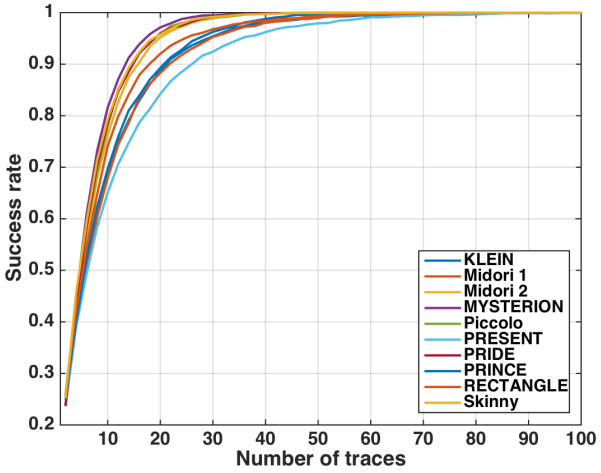

Fig. 30: $\sigma = 1$, SNR = 1 (last round)

moreover, confirm that the last round is equal or more resistant than the first round.

For 8-bit inverse S-boxes we again see that all three ciphers perform nearly equivalently. When comparing to the 4-bit ciphers we see that for equal σ the differences between the most resilient 4-bit ciphers and the 8-bit ciphers even becomes the greater. Thus, we observe that 4-bit ciphers can be much more resilient than 8-bit ciphers, e.g., for $\sigma = 4$ PRESENT requires 400 traces in order to reach a success rate of 0.9, whereas AES, Robin, Zorro require only around 150. Additionally, when considering the same SNR we can observe the same trend. For example, for SNR= $1/16$ the 8-bit ciphers require 300 traces to reach a success rate of 0.9 and PRESENT 400 traces.

3.4 Attacking First and Last Round

Naturally when ciphertext and plaintext are available, an attacker will choose to attack the first or the last round depending on the attackability. As we showed in the previous subsections that the first round (direct S-box computation) is less resistant than the last round (inverse S-

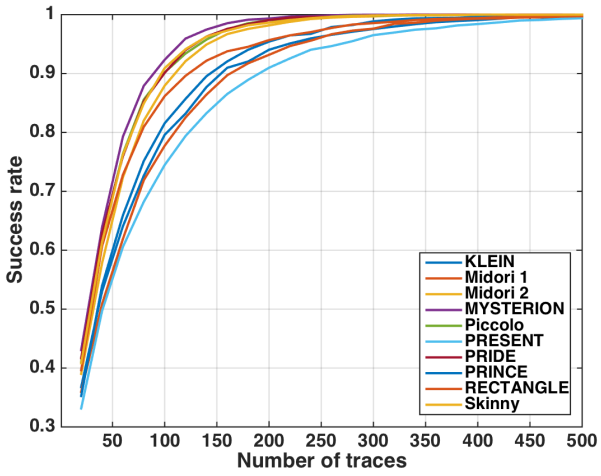

Fig. 31: $\sigma = \sqrt{8}$, SNR = $1/8$ (last round)

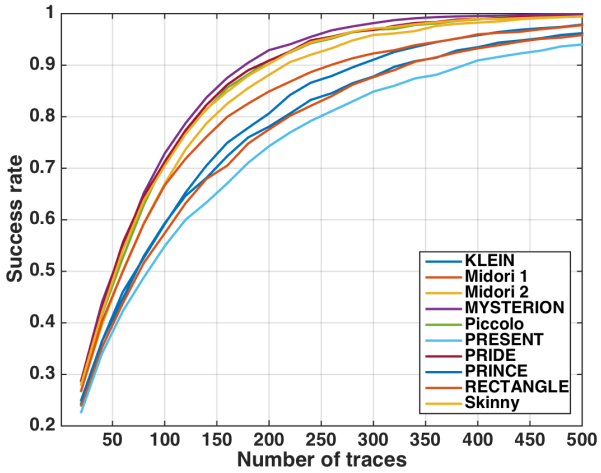

Fig. 32: $\sigma = 4$, SNR = $1/16$ (last round)

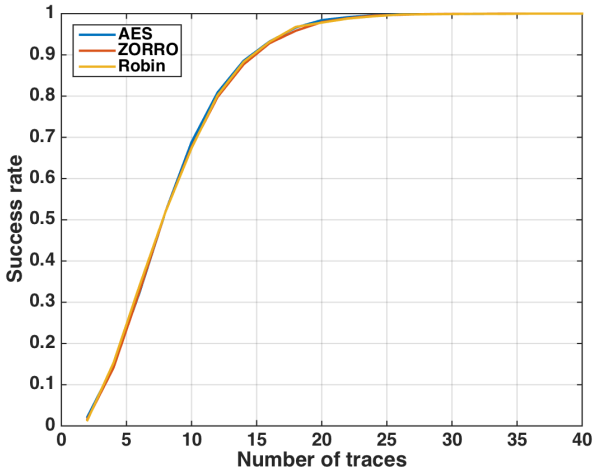

Fig. 33: $\sigma = 1$, SNR = 2 (last round)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

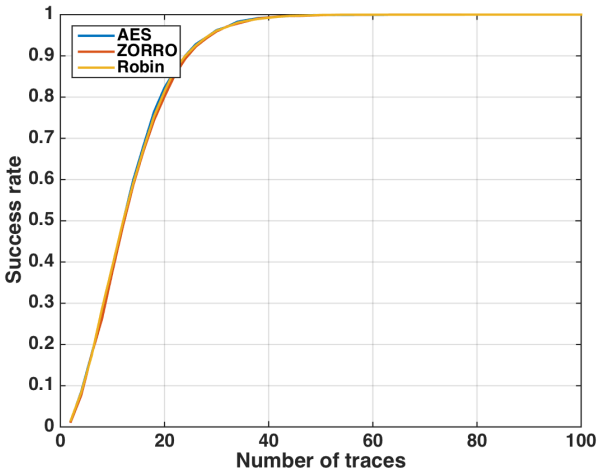

Fig. 34: $\sigma = \sqrt{2}$, SNR = 1 (last round)

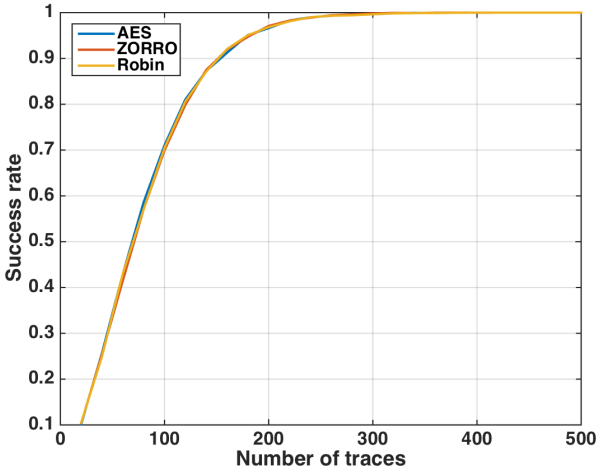

Fig. 35: $\sigma = 4$, SNR = $1/8$ (last round)

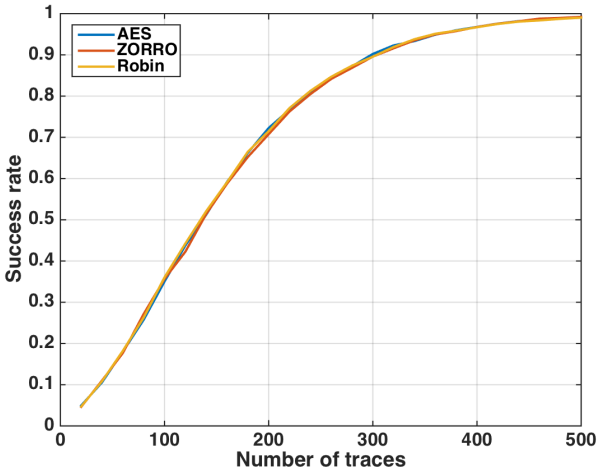

Fig. 36: $\sigma = \sqrt{32}$, SNR = $1/16$ (last round)

box computation). However, in some situations an attacker may even choose to combine the knowlegde gained from the attack on the first and the last round. Most of the investigated ciphers employ a key scheduling algorithm and thus it is fair to assume the the roundkeys from the first and last round are independent. In such a scenario the authors in [27] showed that using the optimal distinguisher on both leakage samples will only bring a benefit compared to two independent attacks on each round if the noise is correlated.

But, the situation differs in case the keys used in the first round and last round can be straightforwardly derived from each other. Loosely speaking, in this case one can take benefit simultaneously from the confusion coefficient of the first and the last round. More precisely, in case of low SNR Eq. (10) (attack on one round) changes to

$$\mathsf{SE} \approx \min_{\mathbf{k} \neq \mathbf{k}^\star} \frac{1}{2(1+\rho^2)} \left(\frac{\alpha_1^2}{\sigma_1^2} \kappa_1(k_1^\star, k) + \frac{\alpha_2^2}{\sigma_2^2} \kappa_2(k_2^\star, k) \right), \quad (11)$$

where $\kappa_1(k_1^\star, k)$ is the confusion coefficient corresponding to the first round and $\kappa_2(k_1^\star, k)$ the confusion coefficient corresponding to the last round.

Accordingly, not the minimum value for each confusion coefficient, but the minimum value of the sum over each value is decisive. This is particularly interesting as we observed that for the S-box and its inverse the distribution differ.

This scenario is observable for LED which does not employ a key scheduling algorithm. Figure 37 plots the confusion coefficient for the LED S-box in black and of its inverse in red. Note that we did not order the confusion coefficients as we are particularly interested in the differences for each key. One can observe that indeed the distribution and the minimum value of both confusion coefficients is not taken for the same key guess, i.e.

$$\arg \min_{k^\star \neq k} \kappa_1(k^\star, k) \neq \arg \min_{k^\star \neq k} \kappa_2(k^\star, k).$$

Thus, taking both rounds into account should be really beneficial from an attackers point of view.

Figure 39 to Fig. 41 show the success rates for the combined attack compared to attacks on the first and last round. Clearly, the attack using the information from both rounds is much more efficient than on the first or on the last round. For $\sigma = 4$ the key can be recovered with a success rate of 0.9 within 100 traces for both rounds, and 400 traces when only considering the last round. As a remark, for 4-bit intermediate states, this example additionally highlights the important role the confusion coefficient (underlying leakage model), and that not only the SNR is a key factor influencing the success rate as assumed in state-of-the-art works.

4 Recovering Intermediate Round States

In the previous section we were interested in the influence of the S-box operation in recovering the (round)key and, in particular, in the relationship between different predicted intermediate states measured by the confusion coefficient. In this section we slightly change our focus as we are interested in the differences of efficiencies between ciphers with 4-bit states and 8-bit states. More precisely, we investigate the accuracy when recovering intermediate states directly,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

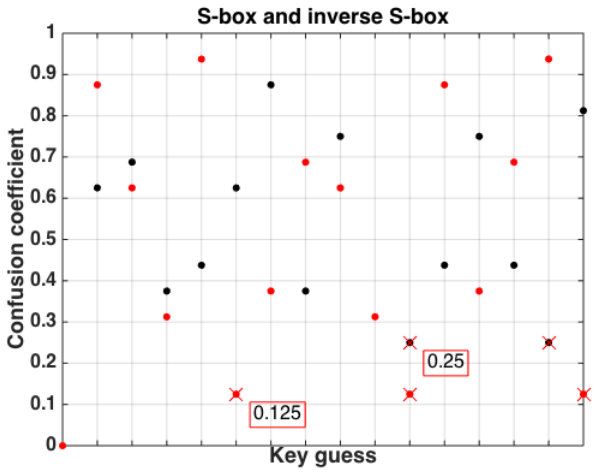

Fig. 37: PRESENT / LED (standard (black), inverse (red))

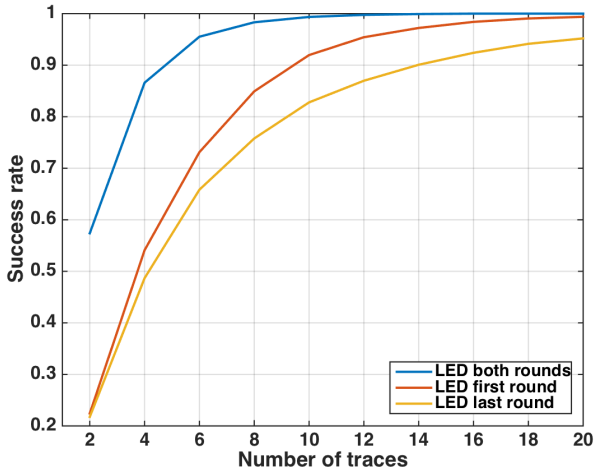

Fig. 38: $\sigma = \sqrt{1/2}$, SNR = 2 (LED)

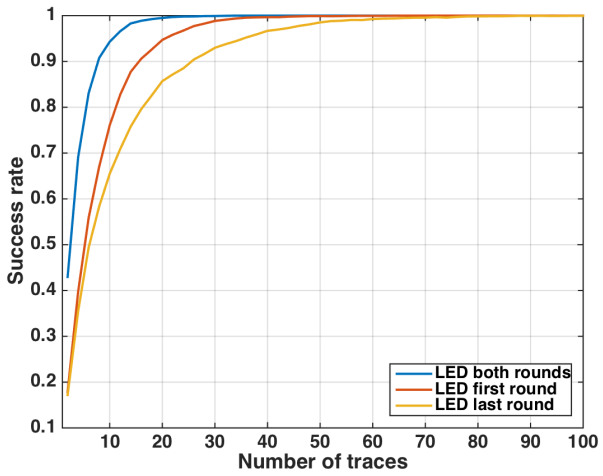

Fig. 39: $\sigma = 1$, SNR = 1 (LED)

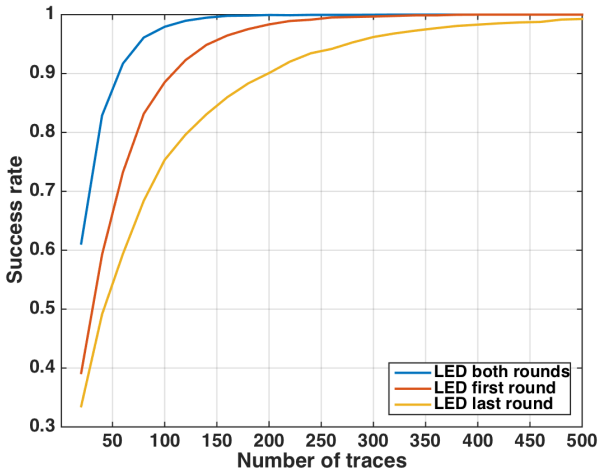

Fig. 40: $\sigma = \sqrt{8}$, SNR = $1/8$ (LED)

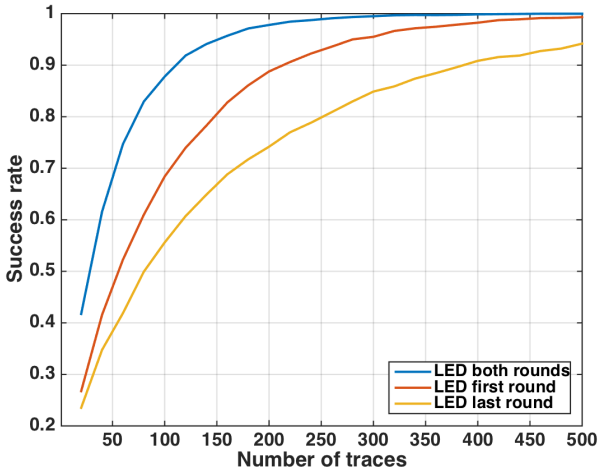

Fig. 41: $\sigma = 4$, SNR = $1/16$ (LED)

where accuracy is the percentage of correctly classification. This scenario is for example of particular interest when considering algebraic side-channel attacks [21]. In this scenario one only has a very limited amount of traces in the attacking phase and is interested in recovering Hamming weight information of intermediate states of ciphers which are then used as inputs in an algebraic system.

Machine learning (ML) is a term encompassing a number of methods that can be used for clustering, classification, regression, feature selection, and other knowledge discovering methods [28]. In supervised machine learning, the algorithm is provided with a set of data instances (i.e., measurements) and data classes (i.e., values of $Y(k^\star)$) in a training phase. The goal of this phase is to "learn" the relationship between the instances and the classes in order to be able to reliably map new instances to the classes in the testing phase.

For our study, we use one algorithm per ML family based on the form in which the output function is represented. In particular, we use Naive Bayes as the simplest algorithm that does not have any parameters to tune. Next,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

from the decision tree family we use the C4.5 algorithm, which is an algorithm considered to be robust to noise. From the perceptron family, we use the Multi Layer Perceptron (MLP) algorithm, which represents an advance over the simple perceptron algorithm.

Our experiments are divided in two phases: training and testing (i.e., attacking) with datasets containing 10 000, 30 000, and 50 000 instances. As common for ML techniques we use 2/3 of the instances for training and 1/3 for testing (e.g., results for 10 000 instances are obtained with 6 650 training instances and 3 350 instances in the testing phase). On the training set we conduct a 10-fold cross-validation with all the considered parameters. Note that the training phase contains a tuning phase in which we select the best parameters for each algorithm. Due to the lack of space, we do not present results from the training phase but we mention the best obtained parameters that are then used in the testing phase. We also conducted the same set of experiments with more advanced ML techniques – Rotation Forest and Support Vector Machines, but the results did not differ significantly from those presented here.

Note that our simulated measurements only contain one feature (time instance), which is commonly accepted for simulated data, but not usual when using ML techniques or profiled SCA (at least before dimension reduction). If one has at his disposal a sufficient number of measurements with many features and the level of noise is low, previous results confirm that such a scenario is easy for profiled attack. However, if the level of noise is high or the number of measurements is too low, then the process becomes more cumbersome. Our study shows that even if only a single feature is available (with sufficient information), the attack can be very powerful. Moreover, with the increase in the number of features, the "curse of dimensionality" can appear: as the number of features grow, the classification effort grows exponentially. Common ways to overcome this problem in SCA are dimension reduction techniques like PCA and LDA. Finally, we note that working with only a single feature also makes theoretical analysis, such as probably approximately correct (PAC) learning, easier; we leave this for future work.

4.1 Naive Bayes (NB)

classifier is a method based on the Bayesian rule (similar to template attacks [29]). Naive Bayes works under the simplifying assumption that the predictor attributes (measurements) are mutually independent among the features given the target class. The existence of highly correlated attributes in a dataset can thus influence the learning process and reduce the number of successful predictions. Additionally, Naive Bayes assumes a normal distribution for predictor attributes and outputs posterior probabilities.

The space complexity for the Naive Bayes algorithm for both the training and the testing phase equals $\mathrm{O}\big(|\mathcal{Y}|Dv\big)$, where $|\mathcal{Y}|$ is the number of classes, D is the number of features, and v is the average number of values for a feature. On the other hand, for the training phase, the time complexity equals $\mathrm{O}\big(QD\big)$ and for the testing phase $\mathrm{O}\big(|\mathcal{Y}|D\big)$, where Q is the number of training examples. Further information about the Naive Bayes algorithm can be found in [30].

4.2 C4.5

is the landmark decision tree algorithm [31]. It is a divide-and-conquer algorithm that splits features at tree nodes using the information-based gain ratio criterion. The node splits in further branches if more information is gained (as measured by the gain ratio) by the split than by keeping all the instances at the node. The runtime of the algorithm is $\mathrm{O}\big(D \times Q \times \log Q\big)$, where D is the number of features and Q is the number of instances [32]. The trees are first grown to full length and pruned afterwards in order to avoid data overfitting.

With the C4.5 algorithm we investigate the influence of the confidence factor parameter that is used for pruning, where smaller values relate to more pruning. We tested that parameter in the range $[0.05, 0.4]$ with a step of 0.05. We conducted a separate tuning phase for each noise level and selected a confidence factor of 0.1 for $\sigma = 1$, 0.2 for $\sigma = 3$, and 0.05 for $\sigma = 5$.

4.3 Multi Layer Perceptron (MLP)

is a feedforward neural network that maps sets of inputs onto sets of appropriate outputs. Multi layer perceptron consists of multiple layers of nodes in a directed graph, where each layer is fully connected to the next one. To train the network, the backpropagation algorithm is used, which is a generalization of the least mean squares algorithm in the linear perceptron. A perceptron is a linear binary classifier applied to the feature vector. Each vector component has an associated weight w_i. Furthermore, each perceptron has a threshold value θ. The output of a perceptron is "1" if the direct sum between the feature vector and the weight vector is larger than zero and "-1" otherwise. A perceptron classifier works only for data that are linearly separable, i.e., if there is some hyperplane that separates all the positive points from all the negative points [28].

MLP must consist of 3 or more layers (since input and output represent two layers) of nonlinearly-activating nodes [33]. We investigate a learning rate parameter in range $[0.05, 0.4]$ with a step of 0.05, a momentum with values $[0.1, 0.2, 0.3, 0.4]$, a training time with values $[400, 500, 600]$, and a validation threshold with values $[10, 20, 30]$. In our experiments we set the number of hidden layers to be equal to $(number\_of\_classes + number\_of\_attributes)/2$, the learning rate is set to 0.1, the momentum applied to the weights during the update is set to 0.2, the training time is set to 500, and the validation threshold to 20.

4.4 4-bit vs. 8-bit

We highlight with a gray cell if the the Area Under Curve (AUC) [34] is close to 0.5 which means that the algorithm is closer to random guessing. Note that in our study we use PRESENT and AES. However, the results (in particular the accuracy) are not specific to these ciphers but rather to the fact of using 4-bit/8-bit S-boxes, the intermediate states and the binomial distribution of the HW.

In addition to the previous scenario of attacking the HW of the output of the S-box, we first perform classifications on key chunks, directly resulting in 16 and 256 classes. The results are presented in Table 2, showing that the accuracy

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

(given in percentages) for PRESENT is higher than for AES for all levels of noise, which seems natural since PRESENT has a significantly smaller number of classes than AES. However, when comparing the best values directly, one can observe that the difference is rather small (e.g., for $\sigma = 1$: 41.55 vs. 38.33). What is interesting to observe, is that the level of noise has much less impact when comparing $\sigma = 3$ and $\sigma = 5$ than when comparing $\sigma = 1$ and $\sigma = 3$. Finally, we observe that the number of measurements does not play a significant role in this case.

Table 4 gives the results for attacking the HW output of the S-box. Again, we observe that the accuracy is higher for PRESENT than for AES, but we notice that for AES the algorithm is rather "randomly" guessing than predicting meaningful classes. This is mainly due to the imbalance of the HWs since they follow a binomial distribution (see Table 3). In particular, for AES with randomly distributed inputs, the HW value 4 is occurring in 27.34% of all events, which is rather high. Therefore, the classifier mainly outputs class 4, giving an accuracy between 27% and 28%. For PRESENT we can see that HW class 2 is occurring in 37.5% of all cases. However, as there are fewer classes in total, the algorithm seems to try to find a reasonable classification.

We additionally investigate the scenario of chosen plaintexts during the profiling phase. Table 5 presents the results for both PRESENT and AES with exactly 1 000 measurements for each class, i.e., the total number of measurements equals 5 000 for PRESENT and 9 000 for AES. We can see that the problem of predicting only a subset of classes is not present and again we observe that classifying PRESENT is more accurate than AES.

5 Conclusions

In this paper, we investigate whether side-channel analysis is easier for lightweight ciphers than e.g. for AES. We cover both profiled and non-profiled techniques where we are interested in recovering secret (round)keys or intermediate states. In the case of non-profiled attacks, we evaluate a number of S-boxes appearing in lightweight ciphers using the confusion coefficient and empirical simulations.

First, we investigate in the scenario where the attacker targets the first round and thus exploits the S-box computation. We observe that the 8-bit S-boxes from AES, Zorro, and Robin perform similarly, whereas for 4-bit S-boxes we have a clear ranking, with the S-box of Piccolo being the weakest to attack and the S-box of KLEIN and Midori (1) the hardest.

Interestingly, when considering the last round and thus the inverse S-box operation the ranking changes such that Myterion is the weakest and PRESENT/LED is the most side-channel resistant cipher from the ones investigated. Moreover, we could observe that attacking the last round is equal or less efficient for all considered ciphers.

Finally, we used the information gained from both rounds together, where this approach is of interest when the cipher does not use round keys from a key scheduling algorithm but rather uses the same (or a straightforward computable) key in each round. LED fulfils this requirement. For a reasonable low SNR, to reach a success rate of 0.9 an attack on both rounds only requires 100 traces, whereas an attack using the first round requires 200 traces and on the last 400 traces. This example highlights the important role the confusion coefficient (relationship between predicted intermediate states under a leakage model from different key hypotheses), and that not only the SNR (even if low) is a key factor influencing the success rate.

Additionally, our result show that we cannot conclude that the 4-bit S-boxes are generally significantly less resistant than the investigated 8-bit S-boxes. In particular, when considering inverse S-boxes we showed that 4-bit S-boxes may be more resistant.

For profiled attacks, we analyze several machine learning techniques to recover 4-bit and 8-bit intermediate states. Our results show that attacking 4-bit is somewhat easier than attacking 8-bit, with the difference mainly stemming from the varying number of classes in one or the other scenario. Still, that difference is not so apparent as one could imagine. Since we work with only a single feature and yet obtain a good accuracy in a number of test scenarios, we are confident (as our experiments also confirm) that adding more features will render classification algorithms even more powerful, which will result in an even higher accuracy.

Finally, we did not consider any countermeasures for the considered lightweight algorithms, since the capacity for adding countermeasures is highly dependent on the environment (which we assume to be much more constrained than in the case of AES). However, our results show that a smart selection of S-boxes results in an inherent resilience (especially for 4-bit S-boxes).

Moreover, we show that in case of highly restricted devices, in which countermeasures on the whole cipher are not practically feasible, a designer may choose to only protect the weakest round (first round) in the cipher to increase the side-channel resistant until a certain limit. Future work may concentrate on finding this trade-off between available resources and security requirements, in particular when considering IoT devices.

Acknowledgements

Part of this work has been funded by the ANR CHIST-ERA project SECODE (*Secure Codes to thwart Cyber-physical Attacks*), by the Croatian Science Foundation under the project IP-2014-09-4882, and in part by the Research Council KU Leuven (C16/15/058) and IOF project EDA-DSE (HB/13/020).

References

[1] Kocher, P.C., Jaffe, J., Jun, B.: Differential Power Analysis. In: Proceedings of CRYPTO'99. Volume 1666 of LNCS., Springer-Verlag (1999) 388–397

[2] Gandolfi, K., Mourtel, C., Olivier, F.: Electromagnetic analysis: Concrete results. In: Proceedings of the Third International Workshop on Cryptographic Hardware and Embedded Systems. CHES '01, London, UK, UK, Springer-Verlag (2001) 251–261

[3] Hospodar, G., Gierlichs, B., De Mulder, E., Verbauwhede, I., Vandewalle, J.: Machine learning in side-channel analysis: a first study. Journal of Cryptographic Engineering **1** (2011) 293–302 10.1007/s13389-011-0023-x.

[4] Lerman, L., Bontempi, G., Markowitch, O.: A machine learning approach against a masked AES - Reaching the limit of side-channel attacks with a learning model. J. Cryptographic Engineering **5**(2) (2015) 123–139

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TC.2017.2757921, IEEE Transactions on Computers

TABLE 2: Testing results for classifying a key chunk (nibble or byte)

PRESENT: 16 classes

	10,000			30,000			50,000		
Algorithm	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$
NB	41.55	19.94	12.06	42.62	18.68	13.86	41.72	18.53	14.04
C4.5	40.73	14.85	11.79	41.88	15.79	12.05	41.9	16.08	12.76
MLP	40.67	19.3	11.15	41.4	18.3	14.15	40.82	18.24	13.85

AES: 256 classes

	10,000			30,000			50,000		
Algorithm	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$
NB	38.33	12.67	7.42	37.43	13.04	8.23	38.84	13.29	8.47
C4.5	34.88	9.67	7.69	35.71	10.94	7.18	36.25	10.98	7.04
MLP	35.21	10.94	7.11	37.27	13	7.85	38.67	13.2	8.05

TABLE 3: Occurrences of Hamming weights in %

HW	0	1	2	3	4	5	6	7	8
4-bit	6.25	25	37.5	25	6.25	–	–	–	–
8-bit	0.39	3.12	10.93	21.87	27.34	21.87	10.93	3.12	0.39

TABLE 4: Testing results for classifying the HW of the S-box output

PRESENT: 5 classes

	10,000			30,000			50,000		
Algorithm	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$
NB	51.27	38.55	37.12	51.17	38.57	37.1	51.04	38.92	37.81
C4.5	50.06	38.82	37.03	51.05	38.16	37.19	50.72	38.73	37.59
MLP	51.27	39.12	37.03	51.07	38.47	37.31	50.57	39	38

AES: 9 classes

	10,000			30,000			50,000		
Algorithm	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$
NB	27.67	27.63	28.18	27.07	27.04	27.52	27.94	27.93	28.04
C4.5	27.76	26.91	27.64	27.07	26.77	27.26	27.94	27.94	28.15
MLP	27.64	27.64	27.21	27.03	27.03	27.47	27.93	27.93	28.33

TABLE 5: Results with 1 000 measurements per class, HW model

	PRESENT (5 classes)			AES (9 classes)		
Algorithm	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$	$\sigma = 1$	$\sigma = 3$	$\sigma = 5$
NB	49.7	30.55	24.97	45.32	21.85	19.19
C4.5	50.73	30.79	24.06	43.67	21.26	19.36
MLP	50.12	29.7	24.18	44.14	21.82	19.02

[5] Daemen, J., Rijmen, V.: The Design of Rijndael: AES - The Advanced Encryption Standard. Springer (2002)

[6] Gong, Z., Nikova, S., Law, Y.W. In: KLEIN: A New Family of Lightweight Block Ciphers. Springer Berlin Heidelberg, Berlin, Heidelberg (2012) 1–18

[7] Banik, S., Bogdanov, A., Isobe, T., Shibutani, K., Hiwatari, H., Akishita, T., Regazzoni, F.: Midori: A Block Cipher for Low Energy (Extended Version). Cryptology ePrint Archive, Report 2015/1142 (2015) http://eprint.iacr.org/.

[8] Journault, A., Standaert, F.X., Varici, K.: Improving the security and efficiency of block ciphers based on ls-designs. Designs, Codes and Cryptography (2016) 1–15

[9] Guo, J., Peyrin, T., Poschmann, A., Robshaw, M.J.B.: The LED Block Cipher. In Preneel, B., Takagi, T., eds.: CHES. Volume 6917 of LNCS., Springer (2011) 326–341

[10] Shibutani, K., Isobe, T., Hiwatari, H., Mitsuda, A., Akishita, T., Shirai, T.: Piccolo: An Ultra-Lightweight Blockcipher. In Preneel, B., Takagi, T., eds.: Cryptographic Hardware and Embedded Systems – CHES 2011: 13th International Workshop, Nara, Japan, 2011. Proceedings, Berlin, Heidelberg, Springer Berlin Heidelberg (2011) 342–357

[11] Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In: CHES. Volume 4727 of LNCS., Springer (September 10-13 2007) 450–466 Vienna, Austria.

[12] Albrecht, M.R., Driessen, B., Kavun, E.B., Leander, G., Paar, C., Yalçın, T. In: Block Ciphers – Focus on the Linear Layer (feat. PRIDE). Springer Berlin Heidelberg, Berlin, Heidelberg (2014) 57–76

[13] Borghoff, J., Canteaut, A., Güneysu, T., Kavun, E.B., Knezevic, M., Knudsen, L.R., Leander, G., Nikov, V., Paar, C., Rechberger, C., Rombouts, P., Thomsen, S., Yalçn, T.: PRINCE : A Low-Latency Block Cipher for Pervasive Computing Applications. In Wang, X., Sako, K., eds.: Advances in Cryptology: ASIACRYPT 2012. Volume 7658 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2012) 208–225

[14] Zhang, W., Bao, Z., Lin, D., Rijmen, V., Yang, B., Verbauwhede, I.: Rectangle: a bit-slice lightweight block cipher suitable for multiple

platforms. Science China Information Sciences **58**(12) (2015) 1–15
[15] Beierle, C., Jean, J., Kölbl, S., Leander, G., Moradi, A., Peyrin, T., Sasaki, Y., Sasdrich, P., Sim, S.M.: The SKINNY Family of Block Ciphers and its Low-Latency Variant MANTIS. Cryptology ePrint Archive, Report 2016/660 (2016) http://eprint.iacr.org/2016/660.
[16] Gérard, B., Grosso, V., Naya-Plasencia, M., Standaert, F.: Block Ciphers That Are Easier to Mask: How Far Can We Go? [35] 383–399
[17] Grosso, V., Leurent, G., Standaert, F.X., Varıcı, K. In: LS-Designs: Bitslice Encryption for Efficient Masked Software Implementations. Springer Berlin Heidelberg, Berlin, Heidelberg (2015) 18–37
[18] Fei, Y., Luo, Q., Ding, A.A.: A Statistical Model for DPA with Novel Algorithmic Confusion Analysis. In Prouff, E., Schaumont, P., eds.: CHES. Volume 7428 of LNCS., Springer (2012) 233–250
[19] Thillard, A., Prouff, E., Roche, T.: Success through Confidence: Evaluating the Effectiveness of a Side-Channel Attack. [35] 21–36
[20] Heuser, A., Rioul, O., Guilley, S.: Good is Not Good Enough — Deriving Optimal Distinguishers from Communication Theory. In Batina, L., Robshaw, M., eds.: CHES. Volume 8731 of Lecture Notes in Computer Science., Springer (2014)
[21] Renauld, M., Standaert, F.X.: Algebraic Side-Channel Attacks. In Bao, F., Yung, M., Lin, D., Jing, J., eds.: Inscrypt. Volume 6151 of Lecture Notes in Computer Science., Springer (2009) 393–410
[22] Annelie Heuser, Stjepan Picek, S.G.N.M.: Side-channel analysis of lightweight ciphers: Does lightweight equal easy? Cryptology ePrint Archive, Report 2017/261 (2017) http://eprint.iacr.org/2017/261.
[23] NIST/ITL/CSD: Advanced Encryption Standard (AES). FIPS PUB 197 (Nov 2001) http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.
[24] Ullrich, M., De Cannière, C., Indesteege, S., Ö., K., Mouha, N., Preneel, B.: Finding Optimal Bitsliced Implementations of 4 4-bit S-Boxes. SKEW 2011 Symmetric Key Encryption Workshop (February 2011)
[25] Guilley, S., Heuser, A., Rioul, O.: A Key to Success - Success Exponents for Side-Channel Distinguishers. In Biryukov, A., Goyal, V., eds.: Progress in Cryptology - INDOCRYPT 2015 - 16th International Conference on Cryptology in India, Bangalore, India, December 6-9, 2015, Proceedings. Volume 9462 of Lecture Notes in Computer Science., Springer (2015) 270–290
[26] Cover, T.M., Thomas, J.A.: Elements of Information Theory. Wiley-Interscience (July 18 2006) ISBN-10: ISBN-10: 0471241954, ISBN-13: 978-0471241959, 2nd edition.
[27] Guilley, S., Lerman, L.: Bivariate attacks and confusion coefficients. Cryptology ePrint Archive, Report 2017/263 (2017) http://eprint.iacr.org/2017/263.
[28] Mitchell, T.M.: Machine Learning. 1 edn. McGraw-Hill, Inc., New York, NY, USA (1997)
[29] Chari, S., Rao, J.R., Rohatgi, P.: Template Attacks. In: CHES. Volume 2523 of LNCS., Springer (August 2002) 13–28 San Francisco Bay (Redwood City), USA.
[30] Friedman, N., Geiger, D., Goldszmidt, M.: Bayesian Network Classifiers. Machine Learning **29**(2) (1997) 131–163
[31] Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1993)
[32] Frank, E., Witten, I.H.: Generating Accurate Rule Sets Without Global Optimization. In Shavlik, J., ed.: Fifteenth International Conference on Machine Learning, Morgan Kaufmann (1998) 144–151
[33] Collobert, R., Bengio, S.: Links Between Perceptrons, MLPs and SVMs. In: Proceedings of the Twenty-first International Conference on Machine Learning. ICML '04, New York, NY, USA, ACM (2004) 23–
[34] Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer Series in Statistics. Springer New York Inc., New York, NY, USA (2001)
[35] Bertoni, G., Coron, J.S., eds.: Cryptographic Hardware and Embedded Systems - CHES 2013 - 15th International Workshop, Santa Barbara, CA, USA, August 20-23, 2013. Proceedings. In Bertoni, G., Coron, J.S., eds.: CHES. Volume 8086 of Lecture Notes in Computer Science., Springer (2013)

Annelie Heuser Annelie Heuser is a researcher of the French National Center for Scientific Research (CNRS) at IRISA, Rennes, France. She was a postdoctoral researcher at TELECOM-ParisTech, where she also obtained her PhD in 2016. Prior to that, she was an associated researcher at the Center for Advanced Security Research Darmstadt (CASED) in Germany. Her main research interests lie in the area of side-channel analysis, machine learning, hardware security, and malware detection/ classification.

Stjepan Piceck Stjepan Picek is assistant professor at TU Delft, The Netherlands in the Cyber Security research group. Prior to that, he was a postdoctoral researcher at MIT Computer Science and Artificial Intelligence Laboratory (CSAIL) and before that at KU Leuven, Belgium. He obtained his PhD in 2015 in computer science for Radboud University, Nijmegen, The Netherlands and University of Zagreb, Croatia. His research interests include machine learning, evolutionary computation, and cryptography.

Sylvain Guilley Sylvain Guilley is Director of the Business Line "Think Ahead" at Secure-IC. Sylvain is also "Ingénieur en Chef des Mines" and Professor at Institut TELECOM-TELECOM ParisTech. He has been conducting researches towards defining provable secure architectures for trusted computing for ten years. Sylvain authored various scientific publications and patents in security and embedded systems. He is a member of the IACR, of the IEEE and of the Cryptarchi club. Sylvain graduated from Ecole Polytechnique (X97), TELECOM-ParisTech (ENST 2002), and got a MSc from ENS / Paris 6 University, a PhD from TELECOM-ParisTech (2007) and an HDR from Paris 7 University (2012).

Nele Mentens Nele Mentens received her master and Ph.D. degree from KU Leuven in 2003 and 2007, respectively. Her Ph.D. focused on secure and efficient coprocessor design for cryptographic applications on FPGAs. Currently, Nele is an associate professor at KU Leuven in the COSIC group at the Electrical Engineering Department (ESAT). Her research interests are in the domains of reconfigurable platforms for security purposes, design automation for cryptographic hardware and security in constrained environments. Nele was a visiting researcher for 3 months at the Ruhr University Bochum in 2013 and at EPFL in 2017. Nele was/is the PI in more than 10 finished and ongoing research projects with national and international funding. She served as a reviewer for many international conferences and journals and was/is part of the program committee of 30+ international conferences. Nele was in the examination board of around 15 Ph.D.s. She was the supervisor of one completed Ph.D. and she currently supervises one post-doctoral researcher and three Ph.D. students. Nele is (co-)author in 70+ publications in international journals, conferences and books.