Combining sources of side-channel information

Christophe Genevey-Metat, Benoît Gérard, Annelie Heuser

Combining sources of side-channel information

Christophe Genevey-Metat[1], Benoît Gérard[2,3], Annelie Heuser[1]

[1] Univ Rennes, Inria, CNRS, IRISA, France
[2] Univ Rennes, CNRS, IRISA, France
[3] Direction Générale de l'Armement

Abstract. A few papers relate that multi-channel consideration can be beneficial for side-channel analysis. However, all were conducted using classical attack techniques. In this work, we propose to explore a multi-channel approach thanks to machine/deep learning. We investigate two kinds of multi-channel combinations. Unlike previous works, we investigate the combination of EM emissions from different locations capturing data-dependent leakage information on the device. Additionally, we consider the combination of the classical leaking signals and a measure of mostly the ambient noise. The knowledge of the ambient noise (due to WiFi, GSM, ...) may help to remove it from a noisy trace. To investigate these multi-channel approaches, we describe one option of how to extend a CNN architecture which takes as input multiple channels. Our results show that multi-channel networks are suitable for side-channel analysis. However, if one channel alone already contains enough exploitable information to reach high effectiveness, naturally, the multi-channel approach cannot improve the performance further.

Keywords: Side-channel analysis, profiled attacks, deep learning, power consumption, electromagnetic emanations, multi-channel, neural networks

1 Introduction

Since the end of the second world war and the breaking of ENIGMA, cryptography has moved to a new age. Modern standards such as RSA, DSA, AES or SHA-3 have been the subject of an extensive cryptanalysis effort to assess their security. Nowadays, breaking a product security by finding a theoretical flaw in the cryptography it uses is highly unlikely (except for proprietary cryptography). On the contrary, finding a bug in some cryptographic code, exploiting a misuse of a cryptographic scheme or leveraging the physical implementation of a cryptographic primitive are attack paths that are frequently used. The former one has been introduced as side-channel attacks by Kocher et al. in [10] where the power consumption of some cryptographic device was monitored to recover the secret key.

Since its introduction at the end of the 90's, the idea of exploiting side-channel information observed from a cryptographic device has grown using different physical quantities such as e.g. time, heat, power, electromagnetic field,

photon emission, sound. While for most of them the attacker needs a physical access to the target device, this threat model is not overlooked. Defense against these attacks can be found in smart-cards, set-top boxes, video games consoles or smartphones for instance.

The main challenge for a designer is to determine whether his device is secure up to the level he targeted. Put another way: for a given attacker power, is it possible to break the security of his product? To do so the designer can send his device to evaluation labs that are in charge of making tests to provide evidences of its security level to a certification body.

For a civilian use, products do not need a very high level of security or at least not for a very long time (the lifetime of a smart-card is not more than five years). At the opposite, defense industry is making security products that will be used for tens of years once deployed. Moreover, while a set-top box will at worst be attacked by a mafia organization, military encryption engines may be challenged by high level state agencies. Thus, in the defense setting, the knowledge of what an attacker can do with a given computational power is even more crucial.

Hopefully, it is possible to mitigate side-channel attacks using different kind of techniques. In a nutshell, they either reduce the sensitive signal level, increase the surrounding noise or complicate the links between the signal and the sensitive data. Moreover, the current trend of increasing the chip complexity (many-cores, caches, speculative execution, ...) while making sometimes new security holes is adding pitfalls to the exploitation of side-channel information.

A new trend in side-channel is to explore machine learning (ML) tools. This tools may help in solving problems as trace misalignment or high dimension data. In this paper we present first experiments on the use of ML tools for combining measures from various sources. More precisely, we explore the opportunity of gathering information of different types of electromagnetic probes, from electromagnetic probes at different positions or by combining a signal and a pure noise measure hoping that it may help in removing the noise in the second signal.

2 State-of-the-Art on machine and deep learning techniques for SCA

The first works using machine learning techniques in side-channel analysis showed that Support Vector Machines (SVM) and Random Forests (RF) are effective profiled side-channel attacks [11, 8]. Indeed, template attacks have been shown to be optimal from an information theoretic point of view [6], however, when the set of measurements in the profiling phase is limited SVM can be more efficient due to the underlying estimation problem [7].

More recently, deep learning techniques have shown to be even more advantageous in several settings. Using the advantages from deep learning in side-channel analysis is becoming a very "fruitful" topic, with newly published works very frequently. Still, of how to use the full potential combined with a deeper understanding of how to use deep learning for side-channel analysis may not have been developed yet.

The first work [13] showed that when an implementation is protected with a masking countermeasures neural networks are able to reveal sensitive key information even without the need of a higher order combination function [15] or an additional step of points of interest selection. Due to the use of convolutions exploitable points of interests are combined and it is possible to perform the attack even without specifying leakage models related to the masking scheme.

Shortly after the introduction of deep learning techniques for side-channel analysis, a database of side-channel measurements (called ASCAD) has been published [16] to facilitate comparable research works in this direction. The database consists of EM measurements of an AES-128 implementation protected with a masking countermeasure. Furthermore, the authors provide a software tool to artificially add a random delay countermeasure. Together with the database the authors provide a study of neural networks architectures and parameter selections.

On the same lines as against masking countermeasures, it has been shown that neural networks are extremely effective against random delay countermeasures [4]. Again, due to the use of convolutions many types of shuffling and random delays are becoming less effective to be used as side-channel protection.

Moreover, to even strengthen profiled side-channel attacks based on neural networks, recent works showed techniques to further improve their attacking strength. The authors in [4] highlighted that data argumentation techniques, i.e. the addition of artificial data, is significantly improving the success of an attack when shuffling (jitter-based) protections are present. In the machine learning domain, and particularly, for deep learning, the problem of overfitting to given data in the learning phase, is a crucial problem. In [9] the authors showed that indeed overfitting may also be a problematic in side-channel analysis, and they showed that adding zero-mean Gaussian noise is helping to generalize the derived model in the training phase. Using this simple trick, which works without adding additional data, the authors could improve the success of the attacks, and stabilize performances over multiple folds.

In previous works deep learning techniques have been treated mostly as black box techniques. In [14] the authors describe a gradient visualization tool that aims to proceed a post-mortem information leakage characterization after the successful training of a neural network. Using their approach it is possible to visualize which points of interests neural networks are utilizing.

A practical parameter selection guide is given in [12], i.e. the author provides some recommendations and practical hints to either enhance the efficiency from an adversary's perspective or to strengthen the resistance of the cryptographic implementations against these attacks from a security developer's perspective. Another realistic real-world study has been performed in [3], and similarly in [5, 19]. The works investigated the scenario when in fact the profiling and the attacking device is different (to some extend), which is relevant in practice, but not always studied in research.

As highlighted, deep learning techniques, and in particular, convolutional neural networks, have been applied, studied, and enhanced for side-channel anal-

ysis. However, there is still potential that has not been explored yet. We describe a novel approach in the next section that models a stronger attacker and therefore provides a more realistic security assessment.

3 Combination of multiple sources for SCA

3.1 Motivation

A limited number of works analysed the advantage of using multiple sources of side-channel information at once and compared these to mono-channel attacks [18, 2, 20]. In particular, the first work on multi-channel attacks compared the use of power and EM leakage in order to determine what channel provides more information than the other one [18]. The comparison was made with different types of template attacks to evaluate the information theoretic impact. It was shown that when combining the power signal with EM emission, one channel improves the weaknesses of the other one. In [2] the authors provided a broader overview of side-channel analysis using multi-channels. The main contribution [2] was to present a new metric in order to select the channels which are suitable for combination and which channels do not require additional knowledge of the key. The authors in [20] show that template attacks using multi-channels (both an EM channel and a power signal) are superior to attacks that use only a single channel (on CMOS device). Interestingly, this does not hold for unprofiled DPA.

These works provided evidence that the consideration of multi-channels can be beneficial for side channel analysis However, all previous studies used traditional techniques (e.g. template attack, DPA). Given the advantages of machine and deep learning techniques in SCA (see Sect. 2) our approach consists of deriving multi-channel machine and deep learning techniques. This allows us to make use of the information provided from multi-channels as well as the advantages given from machine/deep learning techniques. We are extending the approach of multi-channels also to different EM locations. This is detailed in the next subsection.

3.2 Approach

Combining EM from different locations may have various reasons as we detail next. Clearly, with a fine-grained probe the captured EM leakage may be very localized such that the measured side-channel information only reflects particular parts of the processed program or particular hardware (e.g. RAM) during the execution. Thus, combining different localized EM measurements may complement each other and therefore provide additional information. Besides that, another direction is to particularly concentrate not on leakage related to data, but to measure "pure" noise. Therefore we obtain measurement traces capturing exploitable data dependent leakages and one set of measurement traces capturing noise. Having additional noise measurement traces then allows to combine both sources. Another approach, which we leave for future work, is to preprocess

the data dependent leakage measurement traces first. For example, one straightforward preprocessing step constitutes in noise subtraction, i.e. data dependent leakage minus noise measurement trace.

Additionally to measuring multiple EM locations, another approach constitutes in measuring EM locations as well as power consumption. In this work, we purely use EM channels, but we consider the extension to power signal (or any other physical quantity) as future work.

3.3 Multi-channel setup

We were not able to reproduce the exact same setup as the one used for producing the ASCAD dataset but we tried to be the closest as possible to it. The target we used for experiments is a raw AtMega8515 micro-controller on the AVR STK500 platform[4]. We used the same AES-128 encryption than the one from ASCAD which is protected using a masking countermeasure [1], the compiler optimization flag was set to `-O0` but we did not embedded the SOSSE operating system. The chip frequency were set to 3.686MHz.

The measurements were obtained using different Langer near-field EM probes (two RF-B 0,3-3 and one RF-K 7-4) connected to 30dB amplifiers the overall having a bandwidth maximum frequency of 3GHz. The signal were then digitized by an RTO2014 oscilloscope from Rohde & Schwarz having a bandwidth of 1GHz (thus being the limiting element of the chain).

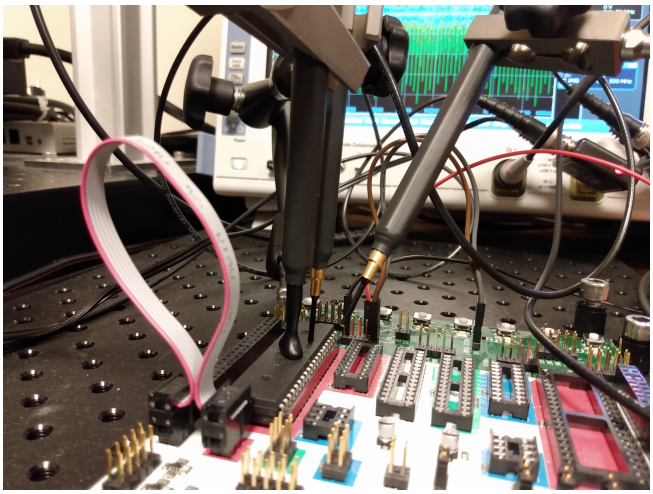


Fig. 1: Multi-channel experiment setup

We focused on the first AES round with a sampling frequency of 1Gs per second and obtained traces containing 20K samples.

[4] For ASCAD a smart-card embedding this micro-controller has been used.

The main two differences between our setup and the one from [16] are

1. the fact that we use a raw micro-controller on a development board instead of a smart-card,
2. the measurement of EM radiations instead of power consumption.

We observed that the leakages we obtained have a different behavior than the one reported in [16], that is different time locations but also different signal powers. However, we obtained similar leakages for the most informative part of he signal (later denoted as P4 see Section 3.5).

3.4 CNN architecture

ASCAD network The first deeply studied neural networks was introduced in [16] and is commonly called ASCAD network. Its network architecture was chosen through exhaustive evaluation of design principles and parameters. The best performing network (from their selection) was relying on the architecture of VGG-16 [17] with 5 blocks and 1 convolutional layer by block, a number of filters equal to (64, 128, 256, 512, 512) with kernel size 11 (same padding), ReLU activation functions and an average pooling layer for each block. The CNN has 2 final dense layers of 4 096 units. In our study we are using this network as a base for extension (see next subsection) and for experiments when only one channel is selected.

Multi-channel ASCAD network In this work, we extend the 1D data representation by another dimension similarly as it is commonly used for images classification with RGB images. Therefore instead of one D input signal, the network has s input layers each consisting of D points. Each input layer s represents the data measurement by one probe. We developed two additional neural networks: for two channels and three channels. The basic parts of the architecture for the three neural networks is kept identical, except the shape of input layer. In particular, the first architecture takes a frame of 700 points of interest (PoI) with one dimension (equal to the measurement of one probe), the second architecture takes a frame of 700 PoI with two dimensions (equal to the measurement of two probes) and the third architecture takes a frame of 700 PoI with three dimensions (equal to the measurement of all probes) as input.

3.5 Experimental results

Metrics To evaluate the amount of leakage, the ability to classify different labels, and to retrieve the key, we use the signal-to-noise-ratio (SNR), the accuracy (acc), the loss, and the guessing entropy (GE). Let X denote the captured EM measurement for one channel, let Y be the label that is determined by the plaintext P and the secret fixed key k^*, then we define the metrics as follows.

– The SNR gives the ratio between the deterministic data-dependent leakage and the remaining noise, i.e.

$$SNR = \frac{\mathrm{Var}(\mathrm{E}(X|Y)}{\mathrm{E}(\mathrm{Var}(X|Y)}, \tag{1}$$

where $\mathrm{E}(\cdot)$ is the expectation and $\mathrm{Var}(\cdot)$ the variance of a random variable.
– The accuracy gives the mean of correctly classified labels. For a set of m measurement traces $x_1, \ldots, x_m$ let $y_1, ..., y_m$ be the corresponding correct label, and $\hat{y}_1, \ldots, \hat{y}_m$ be the label predicted by the classifier, then

$$acc = \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}_{y_i = \hat{y}_i}. \tag{2}$$

– As a loss function $L(y, \hat{y})$, the categorical cross-entropy is used. Let $|Y|$ be the amount of labels and let $\hat{p}_{i,j}$ with $1 \leq i \leq m$ and $1 \leq j \leq |Y|$ be the probability that the classifier is predicting the i-th measurement with the label j, then

$$L = -\sum_{j=0}^{|Y|} \sum_{i=1}^{m} (\mathbb{1}_{y_{i,j}=j)} \cdot \log(\hat{p}_{i,j})). \tag{3}$$

– The guessing entropy gives the ranking of the secret key k^* within a vector of key guesses. In particular, the vector of key guesses $g_{i,1}, \ldots, g_{i,|K|}$ for the ith measurement is calculated by mapping each key guess k to a label j with probability $\hat{p}_{i,j}$ and applying the maximum-likelihood principle over 1 to m. The guessing entropy (aka rank) is then the position of the secret key k^* in the sorted vector of key guess, where the sorted is applied to the probabilities in descending order. In other words, the guessing entropy gives the amount of key guesses an attacker needs to perform before he reveals the secret key. In case his first guess is the secret key GE = 0.

Leakage models In this work, we concentrate on the following mappings (leakage models) between the key and the label:

– In the first model (P4) we target the masked substitution output after the substitution operation (SBox)[5]:

$$y = \mathrm{SBox}(t_b \oplus k) \oplus m_b, \tag{4}$$

with t_b being the plaintext byte 2 and the m being the mask of byte 2.
– additionally we target the masked substitution output (P2)[6]:

$$y = \mathrm{SBox}(t_b \oplus k) \oplus m_{out}, \tag{5}$$

For more details on the leakage models we refer to [16].

[5] This leakage model corresponds to $snr4$ in [16].
[6] This leakage model corresponds to $snr2$ in [16].

Cross-validation In order to have a more reliable estimation of the performance of our neural network with different channels (one, two and three channels), we decide to use cross-validation. K-fold cross-validation is a common model validation tool used in machine learning, which randomly splits the dataset into k folds, i.e.:

1. Split randomly the original samples X and their corresponding labels Y into k subsets.
2. Take $k-2$ sets as training $X_{training}$ and 1 as validation set $X_{validation}$ and 1 as 1 test set X_{test}.
3. Train the model on $X_{training}$.
4. Evaluate the performance of the model with a metric (e.g. accuracy, cross-entropy loss) on $X_{validation}$.
5. Test the performance on X_{test} (e.g. compute guessing entropy).

In this work, we use k-fold cross-validation with k equal to 10.

Evaluation We exemplary show a measurement trace for each of the three channels in Fig. 2. Recall that the probe on channel 1 and 2 (EM1,EM2) are placed to capture data-dependent leakage signals, whereas channel 3 (EM3) is capturing mostly noise. Moreover, EM1 and EM2/EM3 are different types of probes, which explains the different amplitude as well.

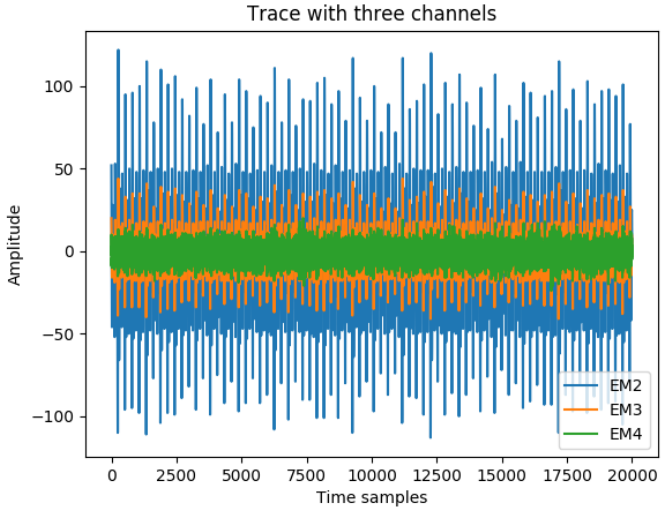
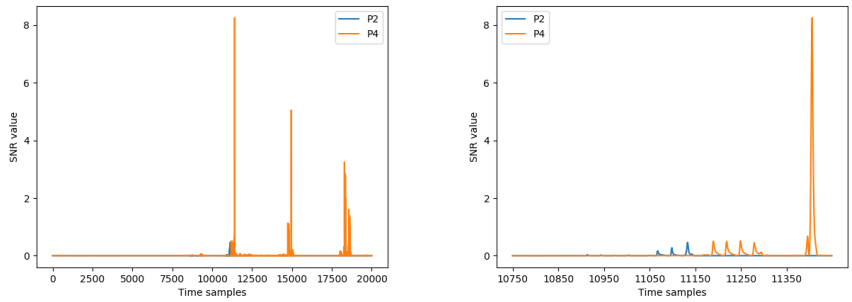


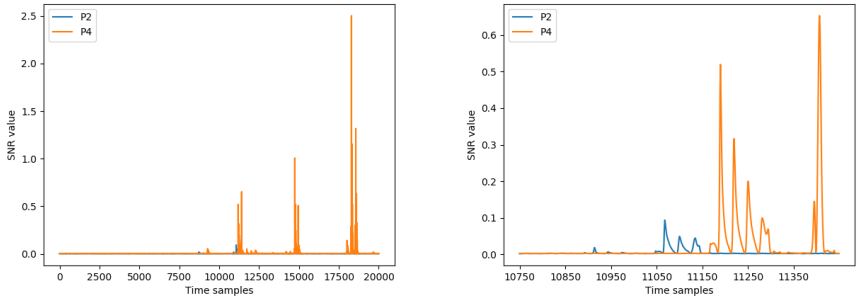
Fig. 2: Measurement trace from each of the three channels

This is confirmed in Fig. 3 showing the SNRs for EM2, EM3, EM4 and the previously defined leakage models. One can see that EM2 provides higher SNRs

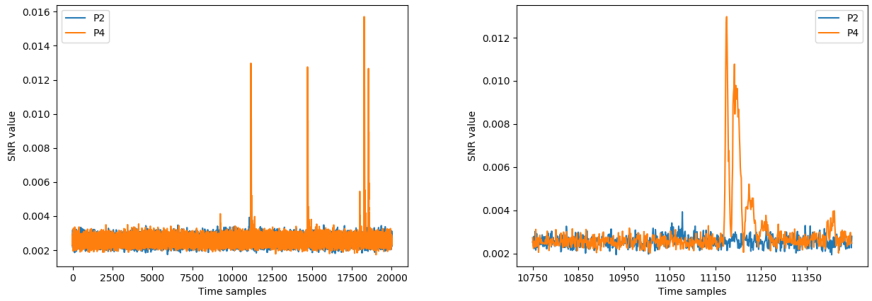
levels then EM3, and EM4, however, the leakage positions in time are consistent. Note that, even though EM4 is very noisy one can still observe minor leakages of P4.



(a) EM2

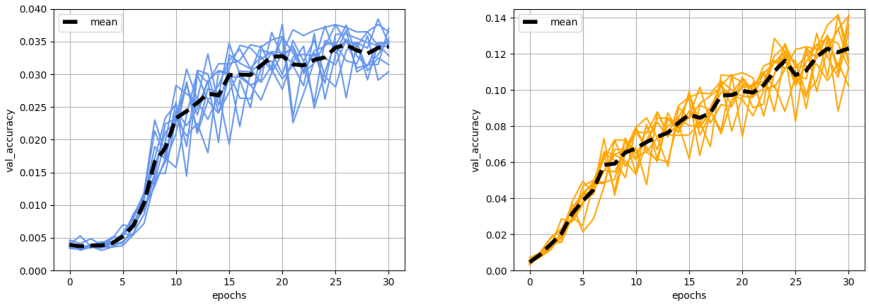


(b) EM3

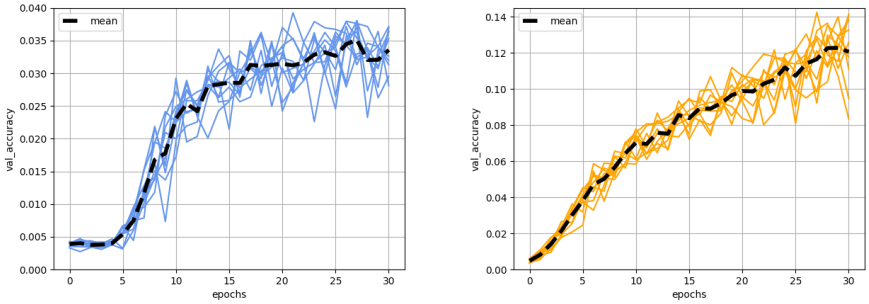


(c) EM4

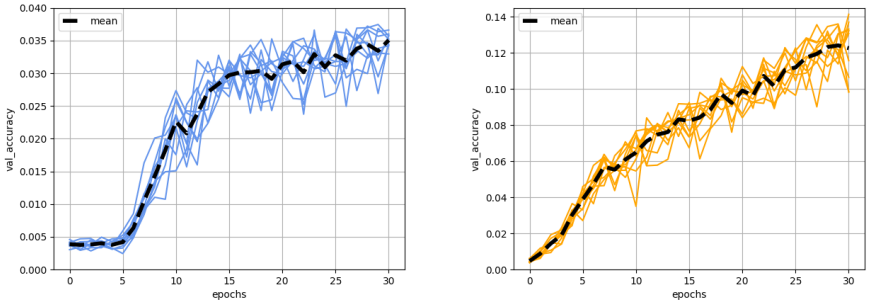
Fig. 3: SNR evaluation for each leakage model, zoom in into considered time frame on the left



(a) 1 channel



(b) 2 channels

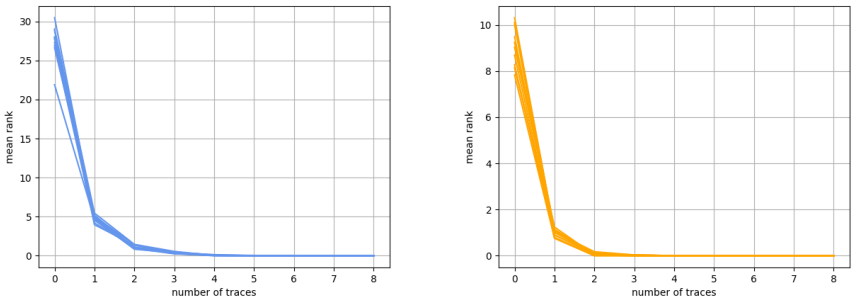


(c) 3 channels

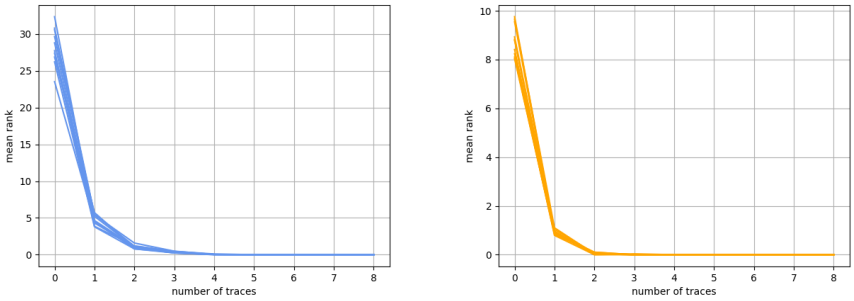
Fig. 4: Accuracy during validation (left: P2, right: P4)

Next, in Figs. 4a-4c we show the accuracy in the validation step for each fold using one, two, and all three channels for each of the leakage models. Additionally, we plot the mean value over all folds in black. In general, one can observe that the validation accuracy for P2 rises quicker than for P4, while P4

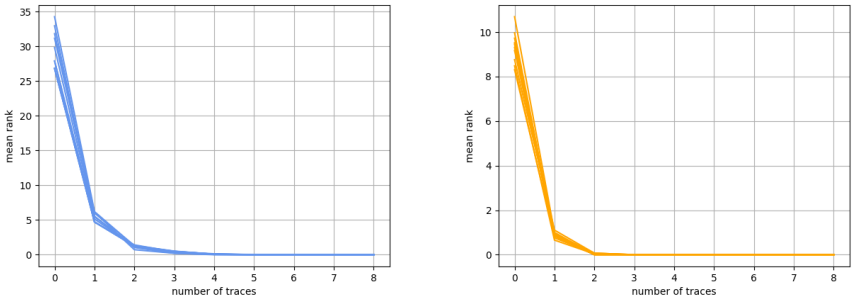
has a higher accuracy value. Further, for both leakage models all channels (one, two or three) are behaving similarly with a very minor advantage using 3 channels. This indicates that the information provided by the first channel is already sufficient in our considered scenario.



(a) 1 channel



(b) 2 channels



(c) 3 channels

Fig. 5: Guessing entropy (rank), left: P2, right: P4

Figure 5 shows the guessing entropy for all three channel options. Like accuracy we see a similar behavior between one, two, and three channels. Leakage model P4 is performing slightly better than P2, which can be explained due to the higher SNR value of P4. Again this indicates that in our considered scenario the information given with one channel is sufficiently high. Indeed, even with one channel the attack is able to achieve a guessing entropy below 5 within only two traces.

4 Conclusion

In this work we motivated the use of multi-source side-channel attacks and its feasibility in this context. We investigated the combination of two data-dependent leakage channels, and the combination with three channels where one channel was mostly capturing ambient noise. Our results show that in our measurement setup using one channel and the previously introduced ASCAD network we are able to reach a guessing entropy below 5 using only 2 traces. Indeed, in this setup the multi-channel approach does not increase the performance further as one channel is already providing enough information. Furthermore, we do not observe a clear different when adding a third channel which mostly captures a noise signal.

We see several potential directions for future work. We would like to study:

- investigation in the scenario where one channel is not yet providing enough information to build an efficient attack model,
- the combination of multiple leakage sources (e.g power consumption with EM),
- analysis of different probe positions,
- different neural network architectures.

References

1. https://github.com/ANSSI-FR/secAES-ATmega8515
2. Agrawal, D., Rao, J.R., Rohatgi, P.: Multi-channel attacks. In: Walter, C.D., Koç, Ç.K., Paar, C. (eds.) CHES 2003. pp. 2–16 (2003)
3. Bhasin, S., Chattopadhyay, A., Heuser, A., Jap, D., Picek, S., Shrivastwa, R.R.: Mind the portability: A warriors guide through realistic profiled side-channel analysis. IACR Cryptology ePrint Archive **2019**, 661 (2019), https://eprint.iacr.org/2019/661
4. Cagli, E., Dumas, C., Prouff, E.: Convolutional neural networks with data augmentation against jitter-based countermeasures - profiling attacks without preprocessing. In: CHES 2017. pp. 45–68 (2017)
5. Das, D., Golder, A., Danial, J., Ghosh, S., Raychowdhury, A., Sen, S.: X-deepsca: Cross-device deep learning side channel attack. In: DAC 2019. pp. 134:1–134:6 (2019)
6. Heuser, A., Rioul, O., Guilley, S.: Good Is Not Good Enough - Deriving Optimal Distinguishers from Communication Theory. In: CHES 2014. pp. 55–74 (2014)

7. Heuser, A., Zohner, M.: Intelligent Machine Homicide - Breaking Cryptographic Devices Using Support Vector Machines. In: Schindler, W., Huss, S.A. (eds.) COSADE. LNCS, vol. 7275, pp. 249–264. Springer (2012)
8. Hospodar, G., Gierlichs, B., Mulder, E.D., Verbauwhede, I., Vandewalle, J.: Machine learning in side-channel analysis: a first study. J. Cryptographic Engineering **1**(4), 293–302 (2011)
9. Kim, J., Picek, S., Heuser, A., Bhasin, S., Hanjalic, A.: Make some noise. unleashing the power of convolutional neural networks for profiled side-channel analysis. IACR TCHES **2019**(3), 148–179 (2019)
10. Kocher, P.C., Jaffe, J., Jun, B.: Differential power analysis. In: CRYPTO '99. pp. 388–397 (1999)
11. Lerman, L., Bontempi, G., Markowitch, O.: Side Channel Attack: an Approach Based on Machine Learning. In: COSADE 2011. pp. 29–41 (2011)
12. Maghrebi, H.: Deep learning based side channel attacks in practice. IACR Cryptology ePrint Archive **2019**, 578 (2019), https://eprint.iacr.org/2019/578
13. Maghrebi, H., Portigliatti, T., Prouff, E.: Breaking cryptographic implementations using deep learning techniques. In: SPACE 2016s. pp. 3–26 (2016)
14. Masure, L., Dumas, C., Prouff, E.: Gradient visualization for general characterization in profiling attacks. In: COSADE 2019. pp. 145–167 (2019)
15. Prouff, E., Rivain, M., Bevan, R.: Statistical analysis of second order differential power analysis. IEEE Trans. Computers **58**(6), 799–811 (2009). https://doi.org/10.1109/TC.2009.15, https://doi.org/10.1109/TC.2009.15
16. Prouff, E., Strullu, R., Benadjila, R., Cagli, E., Dumas, C.: Study of deep learning techniques for side-channel analysis and introduction to ascad database. Cryptology ePrint Archive, Report 2018/053 (2018), https://eprint.iacr.org/2018/053
17. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR **abs/1409.1556** (2014), http://arxiv.org/abs/1409.1556
18. Standaert, F.X., Archambeau, C.: Using subspace-based template attacks to compare and combine power and electromagnetic information leakages. In: Oswald, E., Rohatgi, P. (eds.) CHES 2008. pp. 411–425 (2008)
19. Wang, H., Brisfors, M., Forsmark, S., Dubrova, E.: How diversity affects deep-learning side-channel attacks. Cryptology ePrint Archive, Report 2019/664 (2019), https://eprint.iacr.org/2019/664
20. Yang, W., Zhou, Y., Cao, Y., Zhang, H., Zhang, Q., Wang, H.: Multi-channel fusion attacks. Trans. Info. For. Sec. **12**(8), 1757–1771 (2017)