IET Information Security

Research Article

IET Journals
The Institution of Engineering and Technology

Robust profiled attacks: should the adversary trust the dataset?

ISSN 1751-8709
Received on 7th January 2016
Revised 16th June 2016
Accepted on 23rd July 2016
doi: 10.1049/iet-ifs.2015.0574
www.ietdl.org

Liran Lerman[1] ✉, Zdenek Martinasek[2], Olivier Markowitch[1]

[1]*Quality and Security of Information Systems, Computer Science Department, Université libre de Bruxelles, Belgium*
[2]*Department of Telecommunications, Faculty of Electrical Engineering and Communication, Brno University of Technology, Czech Republic*
✉ *E-mail: llerman@ulb.ac.be*

Abstract: Side-channel attacks provide tools to analyse the degree of resilience of a cryptographic device against adversaries measuring leakages (e.g. power traces) on the target device executing cryptographic algorithms. In 2002, Chari *et al.* introduced template attacks (TA) as the strongest parametric profiled attacks in an information theoretic sense. Few years later, Schindler *et al.* proposed stochastic attacks (representing other parametric profiled attacks) as improved attacks (with respect to TA) when the adversary has information on the data-dependent part of the leakage. Less than ten years later, the machine learning field provided non-parametric profiled attacks especially useful in high dimensionality contexts. In this study, the authors provide new contexts in which profiled attacks based on machine learning outperform conventional parametric profiled attacks: when the set of leakages contains errors or distortions. More precisely, the authors found that (i) profiled attacks based on machine learning remain effective in a wide range of scenarios, and (ii) TA are more sensitive to distortions and errors in the profiling and attacking sets.

1 Introduction

Side-channel attacks analyse physical characteristics (called *leakages* or *traces*) of cryptographic devices related to the execution of the implementation of a cryptographic algorithm. The physical analysis aims to extract a secret value (also known as the sensitive information) such as the secret key. The rationale is that there is a relationship between the manipulated data, the executed operations and the physical properties observed during the execution of the cryptographic device. The physical properties that can be extracted are, for examples, the execution time of a cryptographic algorithm [1], the electromagnetic emanation [2] or the power consumption of the device [3]. From an industrial point of view, side-channel attacks lead to extremely effective and successful attacks against (certified and uncertified) industrial products [4–6].

We focus on side-channel attacks based on the power consumption called Power Analysis (PA) although our analysis can be applied similarly to other physical properties. Power attacks were introduced by Kocher and are generally based on two methods: Simple Power Attacks (SPA) and Differential Power Attacks (DPA) [1]. SPA recover the target value by searching patterns in the measured traces. On the other hand, DPA compare the measured leakages with hypothetical leakages estimated with guessed target values. DPA are the method of choice when the traces have a low signal-to-noise ratio and when the cryptographic algorithm executes the same operation independently of the value of the secret key. We focus henceforth on DPA.

From a different perspective, side-channel attacks can be classified into two categories according to the resources available to the adversary, namely non-profiled and profiled attacks. *Non-profiled attacks* (introduced by Kocher *et al*. [3]) work under the assumption that the adversary has knowledge on the physical behaviour of the cryptographic device (e.g. the power consumption of a device is linearly correlated to the manipulated data). *Profiled attacks*, introduced by Fahn *et al.* [7], extract knowledge (during the *learning phase* also known as the *profiling phase*) about the physical properties of a target cryptographic device from a similar device (called *profiling device*). More precisely, the adversary first extracts a set of leakages (called a *profiling set*) from the profiling device in order to build a model. Afterward, the adversary extracts the target value from a set of leakages (called an *attacking set*) measured on the target device. We focus on profiled attacks introduced as the strongest leakage analysis in an information theoretic sense [8].

Conventional profiled attacks family includes template attacks (TA) [9] and stochastic attacks [10]. In recent years, the cryptographic community has been exploring the potential of profiled attacks based on machine learning models [11–19]. According to the previous papers analysing machine learning based attacks, profiled attacks based on learning models outperform conventional profiled attacks in high dimensionality contexts.

1.1 Our contributions

Several papers highlighted that the characteristics of leakages vary across the measured leakages [20–23]. More precisely, real world datasets often suffer from errors or distortions in the measured leakages that may affect the efficiency of the adversary. The impact of these issues on the success of an attack can be reduced with pre-processing techniques, but cannot be entirely removed [23]. In this paper, we aim to verify which profiled attack (among conventional profiled attacks and profiled attacks based on machine learning) has the lowest sensitivity to modifications of the characteristics of leakages.

1.2 Organisation of this paper

The rest of the paper is organised as follows. Section 2 contains notations, background on side-channel attacks as well as the considered profiled attacks. Section 3 presents our different scenarios, the target algorithm, the experimental setting as well as the results of the robustness of profiled attacks. Eventually, Section 4 concludes the paper and discusses perspectives of future work.

2 Background

2.1 Side-channel attacks

In the following, we use the acronym SNR for the signal-to-noise ratio, and we denote l a leakage measured on a cryptographic device. Let $\boldsymbol{l}_y$ be a leakage measured on a device that manipulates a target value y (also known as label and class). Let l_y^j be the jth measured leakage associated to the target value y. Let $\boldsymbol{l}_y(t)$ be the

tth time sample (also known as a feature) of the leakage trace $\boldsymbol{l}_y$. This sample represents the interesting point of a leakage. We consider contexts where each trace $\boldsymbol{l}_y$ represents a vector of n_s interesting points (samples), i.e.:

$$\boldsymbol{l}_y = \left[\boldsymbol{l}_y(t) \in \mathbb{R} \,|\, t \in [1; n_s]\right]. \tag{1}$$

The profiling set $\mathcal{L}_{\text{PS}}$ (sometimes denoted as a training set or a learning set) represents a set of N_p (profiling) leakages measured on a device under control and similar to the target device. This set of leakages allows during the profiling step to estimate a parameter θ used in the profiled model (denoted $A(\mathcal{L}_{\text{AS}}, \theta)$) that returns, during the attack step, the most probably target value y based on an attacking set $\mathcal{L}_{\text{AS}}$ (that contains attack leakages) obtained by measuring the target device.

Algorithm 1 summarises how a profiled model $A(\cdot, \cdot)$ predicts the target value with a profiling and an attacking sets.

Algorithm 1: How to predict the most likely target value associated to an attacking set.
Require: A profiling set $\mathcal{L}_{\text{PS}}$ and an attacking set $\mathcal{L}_{\text{AS}}$
Ensure: The prediction $\hat{y}$ of a profiled model $A(\cdot, \cdot)$

1. Profiling step:
   (a) Implement the crypto algorithm on a controlled device (similar to the target device)
   (b) Collect a set of profiling leakages for each target value on the controlled device
   (c) Estimate the parameter θ with the profiling set (denoted $\mathcal{L}_{\text{PS}}$)
2. Attack step:
   (a) Collect a set of attack leakages (denoted $\mathcal{L}_{\text{AS}}$) on the target device
   (b) $\hat{y} = A(\mathcal{L}_{\text{AS}}, \hat{\theta})$

2.2 Template attacks (TA)

TA use the profiling set $\mathcal{L}_{\text{PS}}$ in order to estimate a leakage model per target value y denoted as $\hat{\Pr}_{\text{model}}\left[\boldsymbol{l}_y \,|\, \hat{\theta}_y\right]$ where $\hat{\theta}_y$ represents the (estimated) parameters of the leakage probability density function. During the attack step, TA use an attacking set $\mathcal{L}_{\text{AS}}$ and select the target value $\hat{y}$ maximising the product of posterior probabilities:

$$\hat{y} = A(\mathcal{L}_{\text{AS}}, \hat{\theta}) = \underset{y}{\text{argmax}} \prod_{l \in \mathcal{L}_{\text{AS}}} \frac{\hat{\Pr}_{\text{model}}\left[l \,|\, \hat{\theta}_y\right] \cdot \Pr[y]}{\hat{\Pr}_{\text{model}}[l]}, \tag{2}$$

where $\hat{\theta}$ represents the set of parameters. We consider that the parameter $\hat{\theta}_y$ corresponds to the mean vector $\hat{\boldsymbol{\mu}}_y$ and the covariance matrix $\hat{\boldsymbol{\Sigma}}_y$ of the Gaussian (leakage) probability density function associated to the target value y as proposed by the seminal work of Chari *et al.* [8], i.e.:

$$\begin{aligned} &\hat{\Pr}_{\text{model}}\left[l \,|\, \hat{\theta}_y = \{\hat{\mu}_y, \hat{\boldsymbol{\Sigma}}_y\}\right] \\ &= \frac{1}{\sqrt{(2\pi)^{n_s} \det(\hat{\boldsymbol{\Sigma}}_y)}} \mathrm{e}^{-(1/2)(l-\hat{\mu}_y)\hat{\boldsymbol{\Sigma}}_y^{-1}(l-\hat{\mu}_y)^{\top}}, \end{aligned} \tag{3}$$

where $\det(\hat{\boldsymbol{\Sigma}})$ denotes the determinant of the matrix $\hat{\boldsymbol{\Sigma}}$. We call this conventional template attack as classical template attack (CTA) in the following. Furthermore, we consider the efficient template attack (ETA) suggested by Choudary *et al.* [24] in which we pool the covariance matrices across all the target values. In other words, ETA estimates one covariance matrix with all the leakages obtained in the profiling set.

2.3 Support vector machines (SVM)

SVM are the most successful techniques in classification [25]. In a binary classification setting (e.g. $y = 1$ or $y = -1$), if the two classes are separable, SVM compute from the profiling set a separating hyperplane $w^{\top} l + b$ (where w and b are estimated values) allowing to estimate the target value $\hat{y}$ from a leakage l according to the decision rule:

$$\hat{y} = A(l, \theta) = \begin{cases} 1 & w^{\top} l + b > 0 \\ -1 & \text{otherwise} \end{cases}, \tag{4}$$

where $\theta = \{w \in \mathbb{R}^{n_s}, b \in \mathbb{R}\}$, and $\{-1,1\}$ represents the space of target values.

To reduce the error due to noise in the profiling leakages, SVM select the hyperplane with the maximal margin, where the margin is the sum of the distances from the hyperplane to the closest profiling leakages of each of the two classes. Cortes *et al.* [25] show that solving the following convex optimisation problem allows to select the value of w and b that maximise the margin:

$$\min_{w} \frac{1}{2}(w^{\top} w), \tag{5}$$

subject to:

$$y(w^{\top} l_y^j + b) \geq 1 \quad \forall j, y \tag{6}$$

in the case of binary labels $y \in \{-1, 1\}$.

By introducing Lagrange multipliers (denoted by $\alpha_{j,y} \in \mathbb{R}$), Cortes *et al.* show that the convex optimisation problem can be solved with a linear weighted sum of the profiling leakages. As a result, the decision rule becomes:

$$\hat{y} = \begin{cases} 1 & w^{\top} l + b > 0 \Leftrightarrow \left(\sum_{l_y^j \in \mathcal{L}_{\text{PS}}} \alpha_{j,y} \times y \times l_y^j\right)^{\top} l + b > 0 \\ -1 & \text{otherwise} \end{cases} \tag{7}$$

In a compact manner, we write de decision rule as follows:

$$\hat{y} = \begin{cases} 1 & \sum_{l_y^j \in \mathcal{L}_{\text{PS}}} \alpha_{j,y} \times y \times \phi\left(l_y^j, l\right) + b > 0 \\ -1 & \text{otherwise} \end{cases}, \tag{8}$$

where ϕ performs the product of two vectors.

An interesting feature of SVM is that it is possible to adapt the classifier to non-linear classification tasks by performing a non-linear transformation φ of the leakages, the decision rule becomes:

$$\hat{y} = \begin{cases} 1 & \sum_{l_y^j \in \mathcal{L}_{\text{PS}}} \alpha_{j,y} \times y \times \phi\left(\varphi\left(l_y^j\right), \varphi(l)\right) + b > 0 \\ -1 & \text{otherwise} \end{cases} \tag{9}$$

We suppose that $\kappa(\cdot, \cdot)$ (called the kernel function) performs the transformation $\phi(\varphi(\cdot), \varphi(\cdot))$ leading to the decision rule:

$$\hat{y} = \begin{cases} 1 & \sum_{l_y^j \in \mathcal{L}_{\text{PS}}} \alpha_{j,y} \times y \times \kappa\left(l_y^j, l\right) + b > 0 \\ -1 & \text{otherwise} \end{cases} \tag{10}$$

Our experiments considered a radial basis kernel function κ (RBF), which is a commonly encountered solution. The RBF maps the leakages into an infinite dimensional Hilbert space in order to find a hyperplane that efficiently discriminates the leakages. RBF is defined by a meta-parameter γ related to the complexity of the

model. In our experiments, we set γ equal to $1/n_s$, which is a natural choice to compensate the increase of the model complexity due to the increase of the number of points per leakage.

SVM can be generalised to multi-class problems. In our experiments, we considered the '*one-against-all*' approach. In a one-against-all strategy, the adversary builds one binary support vector machine for each target value in order to separate leakages of that target value from leakages of other target values.

2.4 Random forests (RF)

RF represent a set of decision trees (DT). DT are structured as diagrams made of nodes and directed edges, where nodes can be of three types: root (i.e. the top node in the tree), internal and leaf. We consider DT in which (i) the value associated to a leaf is a label, (ii) each edge is associated to a test on the value of a feature, and (iii) each internal node has one incoming edge from a node called the parent node and two outcoming edges to two nodes (called left child and right child).

In the profiling step, the DT generator first associates the whole profiling set to the root. Then the generator splits the set associated to the node in two subsets (called left set and right set) based on a feature that most effectively discriminates the set of leakages associated to different target values. Each subset newly created is associated with a child node: the left set (respectively the right set) is associated to the left child (respectively the right child). The tree generator repeats this process on each derived subset in a recursive manner, until the gain to split the subset is less than some threshold. Eventually, the learning algorithm assigns to each leaf the majority class of leakages in that node. The tree construction may be followed by an additional step called the pruning step in which the DT is simplified by substituting a single leaf in place of a whole sub-tree. In the attack step, the model predicts the label by applying the classification rules (represented by the conditions along the path from the root to a leaf) to the unlabelled leakage to classify.

RF were introduced by Breiman in 2001 to address the problem of instability in large DT, where by instability we denote the sensitivity of a DT structure to small changes in the profiling set (also known as the variance issue) [26]. In order to reduce the variance, RF rely on the principle of models averaging by building a number of DT and returning the most consensual prediction. This means that the predicted output $\hat{y}$ of an attack leakage is calculated through a majority vote of the set of trees.

RF are based on two aspects. First each tree is constructed with a different set of profiling leakages through the boostrapping method. This method builds a profiling set (called a *bootstrap sample*) for each DT by sampling with replacement the original profiling set. Second, each tree is built by adopting a random partitioning criterion. This idea allows to obtain decorrelated trees, thus improving the accuracy of the resulting RF. More precisely, in conventional DT each node is split using the best split among all features. In the case of RF, each node is split using the best among a subset of features randomly chosen at that node. In our experiment, we considered a subset of $\sqrt{n_s}$ features as suggested by James *et al.* [27]. Moreover, unlike conventional DT, the trees of the RF are fully grown and are not pruned. In other words, each leaf contains leakages associated to the same target value. This implies null profiling error but large variance and consequently a small success rate for each single tree. The average of trees represents a remedy to the variance issue, and allows the design of an overall more accurate predictor.

2.5 Multilayer perceptrons (MP)

We use MLP executing basic functions called neurons (also known as perceptrons) that output values between −1 and 1. A neuron generates the output value y by executing the composition of two functions f and g, that is:

$$y = f(g(\boldsymbol{x}, \theta)), \tag{11}$$

where $\boldsymbol{x}$ is the input vector, $f(\cdot)$ is a non-linear function (called the activation function), and $g(\cdot, \theta)$ is a linear function (parameterised by θ) allowing to transform a vector of real numbers to a scalar. Our experiments consider the non-linear weighted sum function, that is:

$$y = f\left(\sigma + \sum_i w(i)\boldsymbol{x}(i)\right), \tag{12}$$

where $\theta = [w, \sigma] = [w(0), w(1), ..., \sigma]$, and $\boldsymbol{x} = [\boldsymbol{x}(0), \boldsymbol{x}(1), ...]$.

The MLP increase the capacity of a neuron by grouping neurons in two or more layers. The connection between the ith neuron and the jth neuron is defined by the weight $w_j(i)$ and σ_j (where $\theta_j = [w_j(0), w_j(1), ..., \sigma_j]$ is the parameter of the jth neuron). The first, the last and the middle layers are called, respectively, input, output and hidden layers.

The input of a neuron in a (hidden or output) layer equals to the weighted output of the neurons associated to the previous layer. In our experiments, the input layer contains n_s neurons (i.e. one input neuron per feature) while the output layer contains Y neurons (where Y is the number of possible target values). As a result, based on one leakage l, each neuron from the input layer (i) manipulates one point in the leakage l, and (ii) forwards the result of the manipulation to the next layer. Eventually, each neuron from the output layer provides a score for each target value associated to the input leakage l, and the predicted value $\hat{y}$ represents the target value having the highest score.

The profiling step adjusts each parameter θ_j to achieve a desired output value. For this, the network of neurons uses a supervised learning technique called the backpropagation algorithm that minimises for each leakage in the profiling set the difference between the target value and the generated target value by the network. For the sake of shortness, we refer to the book of Bishop [28] for a deeper introduction to multilayer perceptron and to [29] for a presentation of MLP in PA.

Our experiments use two-layers neural networks containing 200 neurons in the hidden layer and using the sigmoid function as the non-linear function $f(\cdot)$.

3 Experiments and discussion

3.1 Description of scenarios

We consider a wide range of cases grouped in four scenarios that are listed in the following:

- Scenario 1: we increase the number of leakages from the profiling set associated to wrong target values. This scenario can be the illustration of a problem in the protocol used in order to build the dataset as already seen in the DPA Contest V4.1.
- Scenario 2: we increase the number of misaligned leakages (from the profiling set and/or from the attacking set). Each (temporal) misaligned leakage is randomly time-shifted from 1 to 6 points from the original leakage. This scenario can be due to a dysfunction in the power supply, an unstable clock, a lack of a good trigger signal or due to countermeasures such as frequency changing, voltage changing and random delay interrupts.
- Scenario 3: we increase the level of noise on leakages from the profiling set and from the attacking set. This scenario represents a context in which several devices (executed near the measurement setting) influence the environmental noise.
- Scenario 4: we increase the mean value of the leakages by adding a constant value (called the DC offset) in the profiling set and then (as a new case) in the attacking set. The DC offset can be the result of (i) a difference between the profiling device (used to build the profiling set) and the target device, or (ii) a difference between the acquisition campaign during the profiling and attacking step.

Based on these scenarios, we test the robustness of five (previously presented) profiled attacks.

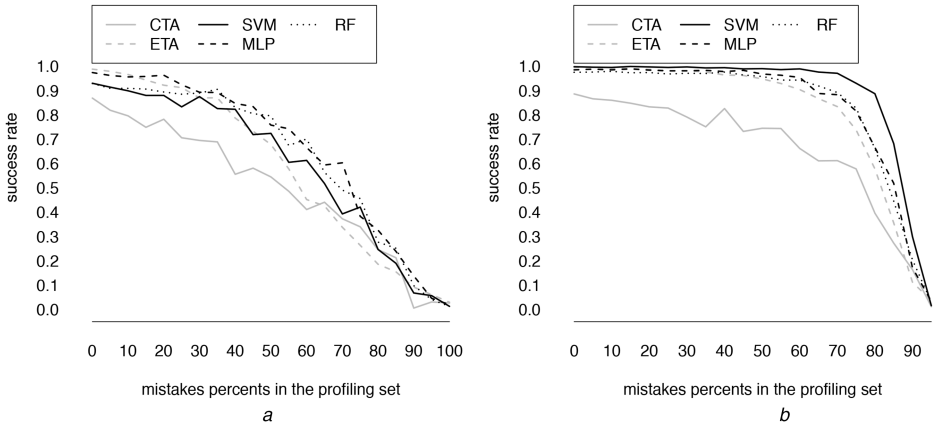

Fig. 1 *Probability to retrieve the target value as a function of the number of mistakes in profiling set for CTA, ETA, SVM, MLP, and RF based on the DPA Contest V4.2*
(a) $N_p = 500$, $n_s = 50$, surroundings = 0 ***(b)*** $N_p = 4000$, $n_s = 50$, surroundings = 0

3.2 Target algorithms

DPA Contest represents an international framework that allows researchers to compare their side-channel attacks under the same conditions. The contest version 4.1 provides leakages associated to the execution of an implementation of AES (128-bit key) protected with a low-entropy Boolean masking scheme called Rotating Sbox Masking (RSM). We refer to the book of Daemen and Rijmen [30] for the interested readers on AES, and to the book of Mangard, Oswald and Popp [31] for an introduction on masking schemes. Regarding RSM, we refer interested readers to the work of Bhasin *et al.* [32].

Few months after the beginning of the DPA Contest V4.1, the organisers provided an improved implementation of RSM (denoted as version 4.2) to avoid most of the identified pitfalls in the previous version. In the following, for the sake of space, we essentially plot the results based on the DPA Contest V4.2. Nevertheless, we obtain the same conclusion based on the dataset provided by the DPA Contest V4.1.

3.3 Description of the testbed

The DPA Contest team used a LeCroy WaveRunner6100A oscilloscope with an EM probe in order to acquire a set of leakages from an 8-bit AVR microcontroller Atmega163. Based on the acquired dataset, we aim to show the sensitivity of profiled attacks by targeting the secret offset of RSM (having an entropy of 4 bits). However, our experiments can be generalised to other sensitive information (e.g. the secret key) and other cryptographic primitives which represent an interesting future work.

To build our datasets based on the set of leakages provided by the DPA Contest, we select the features in the traces that (i) linearly correlate the most with the mask value [Note that an adversary targeting the offset or the mask value leads to the same result in our case: the (Pearson) correlation between them equals to one.], and (ii) are distant each other from at least a certain number of samples (a number denoted as *surroundings* in the following). Note that in the following, we will express the signal-to-noise ratio in decibels (dB).

In each scenario we vary the number of points per leakage (from 20 to 100 points per leakage) denoted n_s, the number of leakages in the profiling set (from 500 to 4000 leakages) denoted N_p, and the surroundings parameter (from 0 to 2). However we provide figures related to the most informative settings for a reason of simplification and space. We use an attacking set that contains 1000 power traces in order to evaluate the quality of attacks. We consider the first order *succes rate* as a metric of comparison (defined as the probability that the model returns the right mask value from one attack leakage).

3.4 Scenario 1: experimental results for mistakes

Fig. 1 shows the probability of each profiled attack to return the target value when varying the percentage of leakages in the profiling set associated to wrong target values. ETA are the method of choice when there is no mistake in the profiling set. CTA provide the worst results overall due to the high number of parameters to estimate leading to a high sensitivity to errors in the profiling set.

It is worth to note that all the methods succeed to have a better success than a random model (i.e. a success rate higher than 1/16) even with more than 80% of mistakes in the profiling set. More precisely, profiled attacks based on machine learning model outperform conventional profiled attacks in the majority of cases (and provide similar results in the other cases) when the percentage of errors is high (especially with the dataset of the DPA Contest V4.2). For example, based on Fig. 1*b*, with 80% of mistakes in the profiling set provided by the DPA Contest V4.2, SVM reach a success rate of 0.887 while the ETA achieve a success rate of 0.578. The rationale of this result is that (i) the increase of mistakes is equivalent to a reduction of the number of leakages in the profiling set leading to be in a high dimensionality context, and (ii) it has been shown that machine learning based attacks outperform TA in a high dimensionality context [16, 18].

3.5 Scenario 2: experimental results for misalignments

Misaligned leakages are easier to exploit (compared with mistakes in the profiling set) since the signal related to target values still persist for several instants. Fig. 2 shows the success rate of each model when varying the percentage of misalignments in the profiling set. ETA still provide the best results when the percentage of misalignments is low. On the contrary, CTA underperform all the profiled attacks. Note also that machine learning based attacks provide a higher success than ETA when increasing the percentage of misalignments. For example, based on the DPA Contest V4.2 and with 80% percentage of misalignments in the profiling set, Fig. 2*b* shows that ETA have a success of 0.32 while SVM, MLP and RF reach a success rate higher than 0.95. However, an increase of the surroundings parameter or of the number of points per leakage allow to increase the success of ETA and, as a result, reduce the sensitivity of TA to misalignments in the profiling set.

Fig. 3 shows the results of attacks when leakages from the attacking set are misaligned. The success rate of each model decreases with the percentage of misaligned leakages in the attacking set. Furthermore, the five models perform similarly.

Fig. 4 shows the results when we vary the percentage of misaligned leakages in the profiling and attacking sets. Three observations can be made: (i) CTA have the worst success rate, (ii) ETA and machine learning models perform similarly on the DPA Contest V4.1, and (iii) machine learning models outperform ETA on the majority of cases based on the DPA Contest V4.2. Regarding the last observation, the success rate of models appears

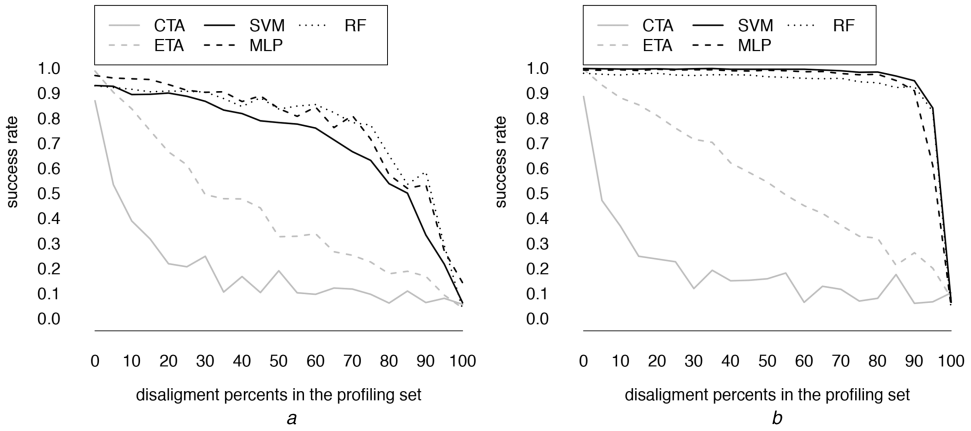

Fig. 2 *Probability to retrieve the target value as a function of the number of misalignments in profiling set for CTA, ETA, SVM, MLP, and RF based on the DPA Contest V4.2*
(a) $N_p = 500$, $n_s = 50$, surroundings = 0 ***(b)*** $N_p = 4000$, $n_s = 50$, surroundings = 0

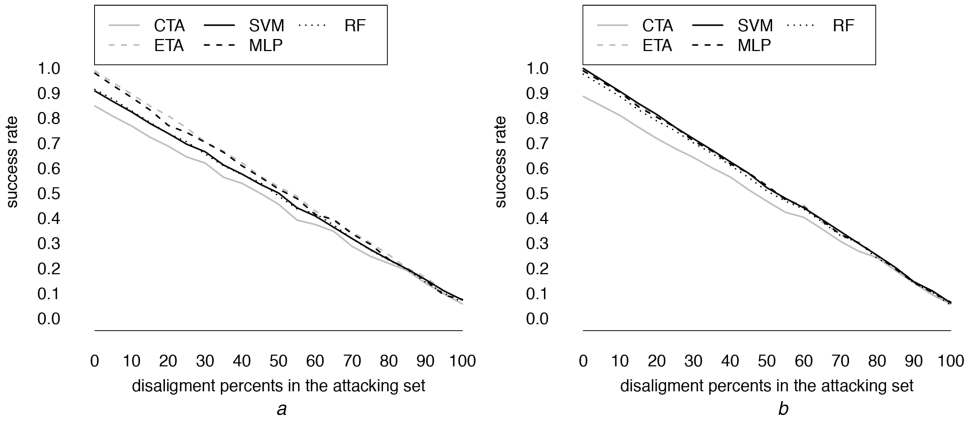

Fig. 3 *Probability to retrieve the target value as a function of the number of misalignments in attacking set for CTA, ETA, SVM, MLP, and RF based on the DPA Contest V4.2*
(a) $N_p = 500$, $n_s = 50$, surroundings = 0 ***(b)*** $N_p = 4000$, $n_s = 50$, surroundings = 0

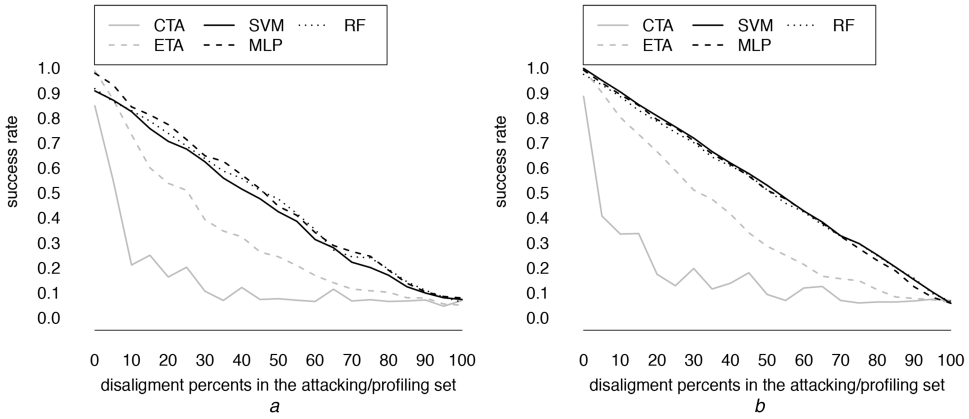

Fig. 4 *Probability to retrieve the target value as a function of the number of misalignments in profiling and attacking sets for CTA, ETA, SVM, MLP, and RF based on the DPA Contest V4.2*
(a) $N_p = 500$, $n_s = 50$, surroundings = 0 ***(b)*** $N_p = 4000$, $n_s = 50$, surroundings = 0

to be related to the sum of (i) the outcomes based on misalignments in the profiling set, and (ii) the results based on misalignments in the attacking set.

3.6 Scenario 3: experimental results for noise

Our third scenario focuses on an increase of the signal-to-noise ratio. Fig. 5 plots the outcomes when varying the signal-to-noise ratio in the profiling set. ETA outperform all the models in low and high signal-to-noise ratio while CTA underperform all the models in a high signal-to-noise ratio.

Fig. 6 shows the results when varying the signal-to-noise ratio in the attacking set. In a low level of noise, ETA outperform or have similar results than machine learning based attacks. In a high level of noise, the models have similar results except SVM that provide the worst result overall.

3.7 Scenario 4: experimental results for DC offset

The last scenario analyses a variation between the profiling and the attacking sets due to a DC offset (i.e. a drift of the global mean of leakages). In a context with a low DC offset applied to the DPA Contest V4.1, ETA provide the best results. However, an increase

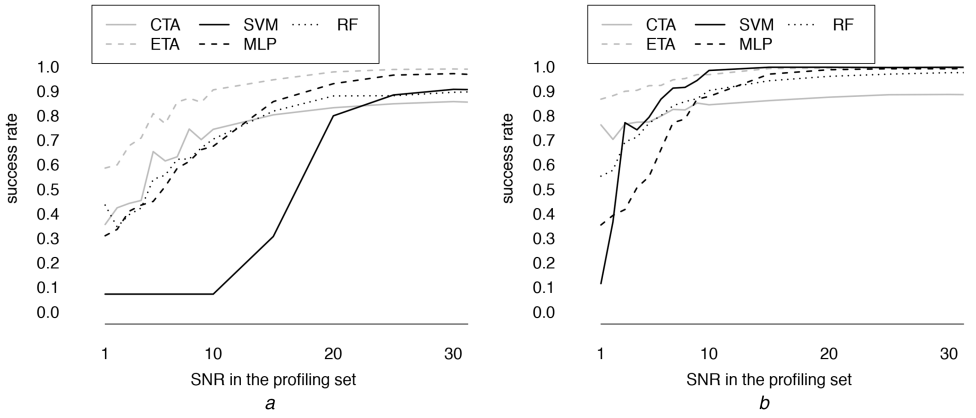

Fig. 5 *Probability to retrieve the target value as a function of the SNR (in dB) in profiling set for CTA, ETA, SVM, MLP, and RF based on the DPA Contest V4.2*

(a) $N_p = 500$, $n_s = 50$, surroundings = 0 ***(b)*** $N_p = 4000$, $n_s = 50$, surroundings = 0

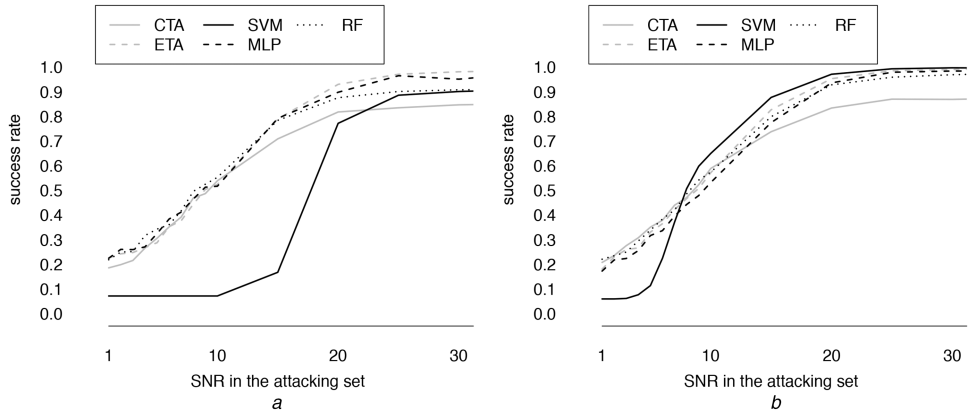

Fig. 6 *Probability to retrieve the target value as a function of the SNR (in dB) in attacking set for CTA, ETA, SVM, MLP, and RF based on the DPA Contest V4.2*

(a) $N_p = 500$, $n_s = 50$, surroundings = 0 ***(b)*** $N_p = 4000$, $n_s = 50$, surroundings = 0

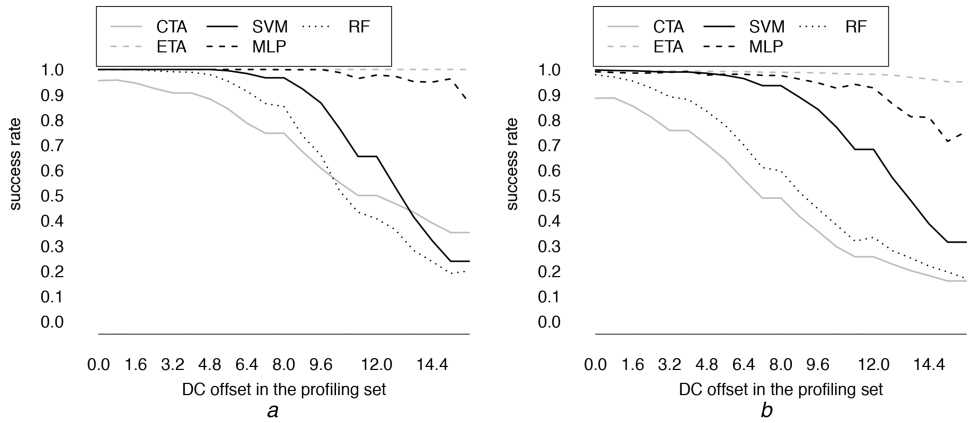

Fig. 7 *Probability to retrieve the target value as a function of the DC offset applied on the leakages from the profiling set for CTA, ETA, SVM, MLP, and RF based on the DPA Contest V4.2*

(a) $N_p = 500$, $n_s = 50$, surroundings = 0 ***(b)*** $N_p = 4000$, $n_s = 50$, surroundings = 0

of the DC offset in the profiling set leads machine learning models (especially a model based on MLP) to outperform ETA.

The results of ETA change when considering the DPA Contest V4.2. Fig. 7 shows the success of attacks when increasing the value of the DC offset in the profiling set using leakages from the DPA Contest V4.2. We obtain similar results when varying the DC offset in the attacking set. ETA reach the best success compared with other models. Note that this result can be due to the fact that the amplitude of the leakages from the DPA Contest V4.2 differs from leakages provided by the DPA Contest V4.1. In other words, these results highlight that ETA outperform machine learning based attacks when the DC offset is low.

3.8 Analysis

Our results are consistent with the no free lunch theorem explaining that the best model for all scenarios does not exist [33]. Nevertheless, the good results of machine learning algorithms compared with (efficient) TA can be explained with the bias-variance theorem recently introduced in the side-channel literature [34].

The bias-variance framework decomposes the error rate (i.e. inversely proportional to the success rate) of an attack in three weighted terms among which the bias and the variance terms. The values of the variance and the bias relate to the attack complexity: a strategy with a high variance means a high sensitivity to the

profiling set while an attack with a high bias indicates a high systematic error compared with the best attack independently of the size of the profiling set.

The bias-variance decomposition shows that (i) CTA have a high variance (i.e. a high sensitivity to the profiling set) due to a high complexity (related to the number of parameters to estimate), (ii) ETA reduce the complexity of TA by reducing the number of estimated parameters, and (iii) machine learning models can vary the variance according to a meta-parameter. For example, SVM compensate the increase of the model complexity due to the increase of the number of points per leakage by reducing the variance term through the modification of the meta-parameter γ. As a result, the learning models can handle a larger error in the profiling set (that increases the complexity to learn) while keeping a lower variance term compared with TA.

4 Conclusion

Our results underline that efficient TA represent the best models when (i) there is no (or a low) variability in the profiling set and in the attacking set, and (ii) the level of noise varies between leakages. Overall, classical TA provide the lowest success to retrieve the target value. However, profiled attacks based on machine learning gain interest for evaluators of cryptographic devices (i) when the number of mistakes (i.e. the number of leakages incorrectly associated to a target value) in the profiling set increases, (ii) when the leakages are misaligned in the profiling and/or attacking sets, and (iii) when the leakages from the profiling set and from the attacking set differ from a high DC offset. In summary, our results are of practical importance for evaluators using tools to analyse the leakages of devices.

Future works include (i) the comparison of the level of robustness of profiled (e.g. TA and profiled attacks based on machine learning) and non-profiled attacks (e.g. the recently model provided by Whitnall *et al.* [23]), and (ii) the exploration of profiled attacks based on other learning models having a lower sensitivity to variation across the leakages.

5 Acknowledgments

The research of L. Lerman is funded by the Brussels Institute for Research and Innovation (Innoviris). Z. Martinasek was financed by the National Sustainability Program under grant LO1401 using the infrastructure of the SIX Center.

6 References

[1] Kocher, P.C.: 'Timing attacks on implementations of diffie-hellman, rsa, dss, and other systems', in Koblitz, N. (ed.): '*Advances in Cryptology – CRYPTO'96, 16th Annual International Cryptology Conference*', Santa Barbara, California, USA, 18–22 August 1996, Proc., Springer, 1996 (LNCS, **1109**), pp. 104–113

[2] Gandolfi, K., Mourtel, C., Olivier, F.: 'Electromagnetic analysis: concrete results', in Koç, Ç.K., Naccache, D., Paar, C. (eds.): '*Cryptographic Hardware and Embedded Systems – CHES 2001, Third International Workshop*', Paris, France, 14–16 May 2001, Proc., Springer, 2001, (LNCS, **2162**) pp. 251–261

[3] Kocher, P.C., Jaffe, J., Jun, B.: 'Differential power analysis', in Wiener, M.J. (ed.): '*Advances in Cryptology – CRYPTO'99, 19th Annual International Cryptology Conference*', Santa Barbara, California, USA, 15–19 August 1999, Proc., Springer, 1999 (LNCS, **1666**), pp. 388–397

[4] Balasch, J., Gierlichs, B., Verdult, R., *et al.*: 'Power analysis of atmel cryptomemory - recovering keys from secure eeproms', in Dunkelman, O. (ed.): '*Topics in Cryptology - CT-RSA 2012 - The Cryptographers' Track at the RSA Conference 2012*', San Francisco, CA, USA, 27 February–2 March 2012. Proc., Springer, 2012 (LNCS, **7178**), pp. 19–34

[5] Oswald, D., Strobel, D., Schellenberg, F., *et al.*: 'When reverse-engineering meets side-channel analysis – digital lockpicking in practice', in Lange, T., Lauter, K.E., Lisonek, P. (eds.): '*Selected Areas in Cryptography - SAC 2013–20th International Conference*', Burnaby, BC, Canada, 14–16 August, 2013, Revised Selected Papers, Springer, 2013 (LNCS, **8282**), pp. 571–588

[6] Zhou, Y., Yu, Yu, Standaert, F.-X., *et al.*: 'On the need of physical security for small embedded devices: a case study with COMP128–1 implementations in SIM cards', in Sadeghi, A.-R. (ed.): '*Financial Cryptography and Data Security – 17th International Conference, FC 2013*', Okinawa, Japan, 1–5 April 2013, Revised Selected Papers, Springer, 2013 (LNCS, **7859**), pp. 230–238

[7] Fahn, P.N., Pearson, P.K.: 'IPA: a new class of power attacks', in Koç, Ç.K., Paar, C. (eds.): '*Cryptographic Hardware and Embedded Systems, First International Workshop, CHES'99*', Worcester, MA, USA, 12–13 August 1999, Proc., Springer, 1999 (LNCS, **1717**), pp. 173–186

[8] Chari, S., Rao, J.R., Rohatgi, P.: 'Template attacks'. In Jr. et al. [9], pp. 13–28

[9] Kaliski, B.S.Jr., Koç, Ç.K., Paar, C. (eds.): 'Cryptographic Hardware and Embedded Systems - CHES 2002'. '*4th International Workshop*', Redwood Shores, CA, USA, 13–15 August 2002, Revised Papers, Springer, 2003 (LNCS, **2523**)

[10] Schindler, W., Lemke, K., Paar, C.: 'A stochastic model for differential side channel cryptanalysis', in Rao, J.R., Sunar, B. (eds.): '*Cryptographic Hardware and Embedded Systems - CHES 2005, 7th International Workshop*', Edinburgh, UK, 29 August–1 September 2005, Proc., Springer, 2005 (LNCS, **3659**), pp. 30–46

[11] Bartkewitz, T., Lemke-Rust, K.: 'Efficient template attacks based on probabilistic multi-class support vector machines', in Mangard, S. (ed.): '*Smart Card Research and Advanced Applications–11th International Conference, CARDIS 2012*', Graz, Austria, 28–30 November 2012, Revised Selected Papers, Springer, 2012 (LNCS, **7771**) pp. 263–276

[12] He, H., Jaffe, J., Zou, L.: *'CS 229 Machine learning - side channel cryptanalysis using machine learning'. Technical Report*, Stanford University, December 2012

[13] Heuser, A., Zohner, M.: 'Intelligent machine homicide - breaking cryptographic devices using support vector machines', in Schindler, W., Huss, S.A. (eds.): '*Constructive Side-Channel Analysis and Secure Design - Third International Workshop, COSADE 2012*', Darmstadt, Germany, 3–4 May 2012. Proc., Springer, 2012 (LNCS, **7275**), pp. 249–264

[14] Hospodar, G., Gierlichs, B., De Mulder, E., *et al.*: 'Machine learning in side-channel analysis: a first study', *J. Cryptographic Eng.*, 2011, **1**, (4), pp. 293–302

[15] Jap, D., Breier, J.: 'Overview of machine learning based side-channel analysis methods'. 2014 14th Int. Symp. on Integrated Circuits (ISIC), December 2014, pp. 38–41

[16] Lerman, L., Bontempi, G., Markowitch, O.: 'Power analysis attack: an approach based on machine learning', *IJACT*, 2014, **3**, (2), pp. 97–115

[17] 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 Eng.*, 2015, **5**, (2), pp. 123–139

[18] Lerman, L., Poussier, R., Bontempi, G., *et al.*: 'Template attacks vs. machine learning revisited (and the curse of dimensionality in side-channel analysis)', in Mangard, S., Poschmann, A.Y. (eds.): '*Constructive Side-Channel Analysis and Secure Design - 6th International Workshop, COSADE 2015*', Berlin, Germany, 13–14 April 2015. Revised Selected Papers, Springer, 2015 (LNCS, **9064**), pp. 20–33

[19] Martinasek, Z., Hajny, J., Malina, L.: 'Optimization of power analysis using neural network'. In Francillon and Rohatgi [35], pp. 94–107.

[20] Choudary, O., Kuhn, M.G.: 'Template attacks on different devices'. In Prouff [36], pp. 179–198

[21] Elaabid, M.A., Guilley, S.: 'Portability of templates', *J. Cryptographic Eng.*, 2012, **2**, (1), pp. 63–74

[22] Renauld, M., Standaert, F.-X., Veyrat-Charvillon, N., *et al.*: 'A formal study of power variability issues and side-channel attacks for nanoscale devices', in Paterson, K.G. (ed.): '*Advances in Cryptology - EUROCRYPT 2011–30th Annual International Conference on the Theory and Applications of Cryptographic Techniques*', Tallinn, Estonia, 15–19 May 2011. Proc., Springer, 2011 (LNCS, **6632**), pp. 109–128

[23] Whitnall, C., Oswald, E.: 'Robust profiling for dpa-style attacks', in Güneysu, T., Handschuh, H. (eds.): '*Cryptographic Hardware and Embedded Systems - CHES 2015–17th International Workshop*', Saint-Malo, France, 13–16 September 2015, Proc., Springer, 2015 (LNCS, **9293**), pp. 3–21

[24] Choudary, O., Kuhn, M.G.: 'Efficient template attacks'. In Francillon and Rohatgi [35], pp. 253–270

[25] Cortes, C., Vapnik, V.: 'Support-vector networks', *Mach. Learn.*, 1995, **20**, (3), pp. 273–297

[26] Breiman, L.: 'Random forests', *Mach. Learn.*, 2001, **45**, (1), pp. 5–32

[27] James, G., Witten, D., Hastie, T., *et al.*: '*An introduction to statistical learning: with applications in R*' Springer Texts in Statistics (Springer, New York, 2014)

[28] Bishop, C.M.: '*neural networks for pattern recognition*' (Oxford University Press, Inc., New York, NY, USA, 1995)

[29] Martinasek, Z., Malina, L., Trasy, K.: '*Profiling power analysis attack based on multi-layer perceptron network*' (Springer International Publishing, Cham, 2015), pp. 317–339

[30] Daemen, J., Rijmen, V.: '*The design of Rijndael: AES - the advanced encryption standard. Information Security and Cryptography*' (Springer, 2002)

[31] Mangard, S., Oswald, E., Popp, T.: '*Power analysis attacks - revealing the secrets of smart cards*' (Springer, 2007)

[32] Bhasin, S., Danger, J.-L., Guilley, S., *et al.*: 'A low-entropy first-degree secure provable masking scheme for resource-constrained devices'. Proc. of the Workshop on Embedded Systems Security, WESS 2013, Montreal, Quebec, Canada, 29 September–4 October 2013, pp. 7:1–7:10

[33] Wolpert, D., Macready, W.G.: 'No free lunch theorems for optimization', *IEEE Trans. Evol. Comput.*, 1997, **1**, (1), pp. 67–82

[34] Lerman, L., Bontempi, G., Markowitch, O.: 'The bias-variance decomposition in profiled attacks', *J. Cryptographic Eng.*, 2015, **5**, (4), pp. 255–267

[35] Francillon, A., Rohatgi, P. (eds.): 'Smart card research and advanced applications'. '*12th International Conference, CARDIS 2013*', Berlin, Germany, 27–29 November 2013. Revised Selected Papers, Springer, 2014 (LNCS, **8419**)

[36] Prouff, E. (ed.): 'Constructive side-channel analysis and secure design'. '*5th International Workshop, COSADE 2014*', Paris, France, 13–15 April 2014. Revised Selected Papers, Springer, 2014 (LNCS, **8622**)