Institut
Mines-Télécom

Template Attacks with Partial Profiles and Dirichlet Priors: Application to Timing Attacks

June 18, 2016

Eloi de Chérisey, Sylvain Guilley, Darshana Jayasinghe and Olivier Rioul

Contents

Motivations

- Derive optimal distinguisher in **timing attacks**.
- Find ways to avoid the **Empty bin Issue**.
- Derive more appropriate distinguishers.
- Compare these distinguishers with **simuations and real attacks.**

Template Attacks

Template side-channel attacks

- Attacker computes distinguisher values using *all the available data*
- A profiling stage is very useful to provide some *a priori* information about the leakage model.
- However, profiling is essentially empirical and may not be exhaustive.
- Therefore, during the attack, the attacker may come up on previously unseen data, which can be troublesome.

State-of-the-art on Template Attacks

TABLE: State-of-the-art on profiled timing attacks

Profiling method	Reference articles
Moments	[Ber05, RM12, WHS12, BRM12]
Distributions	Our paper (Caution about *empty bins*)

Model

- t is the text (plaintext / ciphertext)
- k is the key (k^* is the correct key)
- x is the leakage (time)

$$x_i = \psi(t_i \oplus k^*) \qquad (i = 1, 2, \ldots, q) \tag{1}$$

where $\oplus$ is the XOR (exclusive or) operator and ψ is an unknown function which may contain noise, masking and other hidden parameters[1].

1. The AES meets the secret and the text byte through a xor executed in a fixed number of clock cycles. However, the rest of the AES meets tables and other repositories which are difficult to model and need different amounts of time, hence the use of an unknown function ψ.

Notations

- Hat " $\hat{\cdot}$ " for profiling
- Tilde " $\tilde{\cdot}$ " for online

Definition : number of occurrences

$$\hat{n}_{x,t} = \sum_{i=1}^{\hat{q}} \mathbb{1}_{\hat{x}_i = x, \hat{t}_i = t} \qquad \hat{n}_x = \sum_{i=1}^{\hat{q}} \mathbb{1}_{\hat{x}_i = x},$$

$$\tilde{n}_{x,t} = \sum_{i=1}^{\tilde{q}} \mathbb{1}_{\tilde{x}_i = x, \tilde{t}_i = t} \qquad \tilde{n}_x = \sum_{i=1}^{\tilde{q}} \mathbb{1}_{\tilde{x}_i = x}.$$

Notations

- Hat " $\hat{\cdot}$ " for profiling
- Tilde " $\tilde{\cdot}$ " for online

Definition : probabilities

$$\hat{\mathbb{P}}(x,t) = \frac{1}{\hat{q}} \sum_{i=1}^{\hat{q}} \mathbb{1}_{\hat{x}_i = x, \hat{t}_i = t} = \frac{\hat{n}_{x,t}}{\hat{q}} \qquad \hat{\mathbb{P}}(x) = \frac{1}{\hat{q}} \sum_{i=1}^{\hat{q}} \mathbb{1}_{\hat{x}_i = x} = \frac{\hat{n}_{x}}{\hat{q}},$$

$$\tilde{\mathbb{P}}(x,t) = \frac{1}{\tilde{q}} \sum_{i=1}^{\tilde{q}} \mathbb{1}_{\tilde{x}_i = x, \tilde{t}_i = t} = \frac{\tilde{n}_{x,t}}{\tilde{q}} \qquad \tilde{\mathbb{P}}(x) = \frac{1}{\tilde{q}} \sum_{i=1}^{\tilde{q}} \mathbb{1}_{\tilde{x}_i = x} = \frac{\tilde{n}_{x}}{\tilde{q}}.$$

Contents

Definition (Success Rate)

The success rate SR is probability, averaged over all possible keys, of obtaining the correct key.

$$\mathsf{SR} = \frac{1}{2^n} \sum_{k^*=0}^{2^n-1} \mathbb{P}_{k^*}(\tilde{k} = k^*), \tag{2}$$

where $\tilde{k}$ is the key guess obtained by the distinguisher during the attack.

Optimal attacks

It has been proven [HRG14, Theorem 1, equation (3)] that for equiprobable keys the optimal distinguisher maximizes likelihood :

$$\mathcal{D}_{\mathsf{Optimal}}(\tilde{\mathbf{x}}, \tilde{\mathbf{t}}) = \arg\max_{k \in \mathcal{K}} \mathbb{P}(\tilde{\mathbf{x}} | \tilde{\mathbf{t}} \oplus k). \tag{3}$$

Optimal attacks *versus* Template attacks

In real life, however, the attacker does not know the leakage model perfectly and thus $\mathbb{P}(\tilde{\mathbf{x}}|\tilde{\mathbf{t}} \oplus k)$ is not available. In order to get an estimation of $\mathbb{P}$, we use the profiling data to build $\hat{\mathbb{P}}$. This is the classical *template attack*. The distinguisher becomes

$$\mathcal{D}_{\text{Template}}(\tilde{\mathbf{x}}, \tilde{\mathbf{t}}) = \arg\max_{k \in \mathcal{K}} \hat{\mathbb{P}}(\tilde{\mathbf{x}}|\tilde{\mathbf{t}} \oplus k). \tag{4}$$

This distinguisher is no longer optimal as it does not use the real distribution $\mathbb{P}$. However, if profiling tends to exhaustivity, $\hat{\mathbb{P}}$ and $\mathbb{P}$ will be very close since by the law of large numbers,

$$\forall x, t \quad \hat{\mathbb{P}}(x, t) \underset{\hat{q} \to \infty}{\longrightarrow} \mathbb{P}(x, t).$$

In practice, it is convenient to use the logarithm

$$\arg\max_{k \in \mathcal{K}} \log \hat{\mathbb{P}}(\tilde{\mathbf{x}}|\tilde{\mathbf{t}} \oplus k).$$

In fact, since the samples are i.i.d., we have

$$\hat{\mathbb{P}}(\tilde{\mathbf{x}}|\tilde{\mathbf{t}} \oplus k) = \prod_{i=1}^{\tilde{q}} \hat{\mathbb{P}}(\tilde{x}_i|\tilde{t}_i \oplus k).$$

Therefore, the attacker computes

$$\boxed{\mathcal{D}_{\text{Template}}(\tilde{\mathbf{x}}, \tilde{\mathbf{t}}) = \arg\max_{k \in \mathcal{K}} \sum_{i=1}^{\tilde{q}} \log \hat{\mathbb{P}}(\tilde{x}_i|\tilde{t}_i \oplus k)} \quad (5)$$

where the logarithm is used to transform products into sums for a more reliable computation.

However, we would like to avoid empty bins for which $\hat{\mathbb{P}}(\tilde{x}_i|\tilde{t}_i \oplus k) = 0$, since otherwise, Equation (5) would not be well defined.

Example of Empty Bins

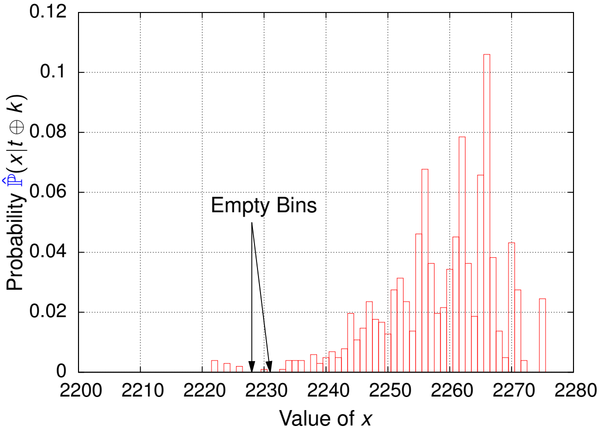


FIGURE: Empirical probability $\hat{\mathbb{P}}(x|t \oplus k)$ for $t = 0$ and $k = 67$

Example of Empty Bins

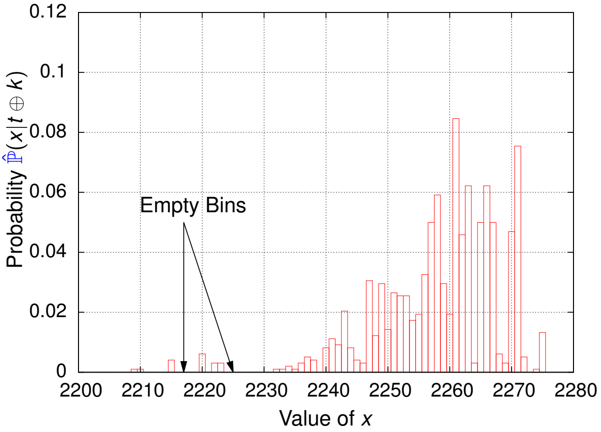


FIGURE: Empirical probability $\hat{\mathbb{P}}(x|t \oplus k)$ for $t = 0$ and $k = 149$

Dirichlet prior

Let some values $\alpha_{x,t} > 0$ and $\alpha = \sum_{x,t} \alpha_{x,t}$.

New distribution :

$$\bar{\mathbb{P}}_\alpha(x,t) = \mathbb{P}(x,t|\hat{\mathbf{x}},\tilde{\mathbf{x}},\hat{\mathbf{t}},\tilde{\mathbf{t}}) = \frac{\hat{n}_{x,t} + \tilde{n}_{x,t} + \alpha}{\hat{q} + \tilde{q} + \sum_{x,t} \alpha_{x,t}}. \tag{6}$$

It is important to notice that for all $(x,t) \in \mathcal{X} \times \mathcal{T}$, one has $\bar{\mathbb{P}}_\alpha(x,t) > 0$.

Learnt MIA

Since our function ψ is unknown, we can create a first-order model $\hat{\psi}$ with the profiled data as

$$\hat{\psi}(t \oplus \hat{k}^*) = \mathrm{Step}\left(\frac{1}{\hat{n}_t} \sum_{i \text{ s.t. } \hat{t}_i = t} \hat{x}_i\right) \qquad (\forall t \in \mathcal{T}). \tag{7}$$

The Step function is a function that ensures the non-injectivity of the model. The simplest way to define Step would be the following :

$$\mathrm{Step}(x) = \frac{\lfloor d \cdot x \rfloor}{d} \qquad (x \in \mathbb{R})$$

where $d > 0$—the greater d, the smaller the step size. This parameter d has to be small enough in order to make the model non-injective [PR09]. With such a model, it is possible to compute a MIA which successfully distinguishes the correct key.

1- Hard drop distinguisher

Definition (Hard Drop Distinguisher)

The hard drop distinguisher is defined as followed :

$$\mathcal{D}_{\text{Hard}}(\tilde{\mathbf{x}}, \tilde{\mathbf{t}}) = \arg\max_{k \in \mathcal{K}} \sum_{i \in \mathcal{I}} \log \hat{\mathbb{P}}(\tilde{x}_i | \tilde{t}_i \oplus k), \tag{8}$$

where $\mathcal{I}$ is defined as

$$\mathcal{I} = \left\{ i \in \{1, \ldots, \tilde{q}\} \mid \forall k \in \mathcal{K},\ \hat{\mathbb{P}}(\tilde{x}_i | \tilde{t}_i \oplus k) > 0 \right\}. \tag{9}$$

2- Soft drop distinguisher

Definition (Soft Drop Distinguisher)

We define the Soft Drop Distinguisher as

$$\mathcal{D}_{\text{Soft}}(\tilde{\mathbf{x}}, \tilde{\mathbf{t}}) = \arg\max_{k \in \mathcal{K}} \sum_{i \text{ s.t. } \hat{\mathbb{P}}(\tilde{x}_i | \tilde{t}_i \oplus k) > 0} \log \hat{\mathbb{P}}(\tilde{x}_i | \tilde{t}_i \oplus k) \;+ \sum_{i \text{ s.t. } \hat{\mathbb{P}}(\tilde{x}_i | \tilde{t}_i, k) = 0} \log \gamma, \quad (10)$$

where $\gamma \in \mathbb{R}_+^*$ is a constant such that $\forall i, k \in \{1, \ldots, \tilde{q}\} \times \mathcal{K}, \quad \gamma \leq \hat{\mathbb{P}}(\tilde{x}_i | \tilde{t}_i \oplus k)$. This means that we penalize data with zero probability. The smaller γ, the harder the penalty.

3- Dirichlet Distinguisher

Definition (The Dirichlet Distinguisher)

We define the Dirichlet Distinguisher as :

$$\mathcal{D}_{\text{Dirichlet}}(\tilde{\mathbf{x}}, \tilde{\mathbf{t}}) = \arg\max_{k \in \mathcal{K}} \bar{\mathbb{P}}_{\alpha}(\tilde{\mathbf{x}} | \tilde{\mathbf{t}} \oplus k). \tag{11}$$

4- Offline-Online Profiled (OOP)

$$\lim_{\alpha \to 0} \bar{\mathbb{P}}_\alpha(x|t) = \frac{\hat{n}_{x,t} + \tilde{n}_{x,t}}{\hat{n}_t + \tilde{n}_t}.$$

This distribution can be denoted as $\bar{\mathbb{P}}_0(x|t)$ and resembles a profiling stage that would start offline and continue online.

Definition (Offline-Online Profiling)

The Offline-Online Profiled (OOP) distinguisher is defined as :

$$\mathcal{D}_{\mathrm{OOP}}(\tilde{\mathbf{x}}, \tilde{\mathbf{t}}) = \arg\max_{k \in \mathcal{K}} \bar{\mathbb{P}}_0(\tilde{\mathbf{x}}|\tilde{\mathbf{t}} \oplus k) \tag{12}$$

5- Learned MIA

Definition (The Learned MIA Distinguisher)

The Learned MIA Distinguisher is defined as :

$$\mathcal{D}_{\text{MIA\_Learned}} = \arg\max_{k \in \mathcal{K}} \tilde{\mathrm{I}}\left(\tilde{\mathbf{x}}; \hat{\psi}(\tilde{\mathbf{t}} \oplus k)\right), \tag{13}$$

where $\tilde{\mathrm{I}}$ is the empirical mutual information [GBTP08].

6- Empty Bin Distinguisher

Definition

The Empty Bin Distinguisher is defined as :

$$\mathcal{D}_{\text{Empty\_Bin}}(\tilde{\mathbf{x}}, \tilde{\mathbf{t}}) = \arg\min_{k \in \mathcal{K}} \sum_{i=1}^{\tilde{q}} \mathbb{1}_{\hat{\mathbb{P}}(\tilde{x}_i | \tilde{t}_i \oplus k) = 0}. \tag{14}$$

Contents

Simulated Model

We test the previous distinguishers upon simulations.

Leakage Model

We use the following leakage model :

$$\forall i \quad x_i = \mathrm{H_w}(\mathrm{SubBytes}(t_i \oplus k^*)) + n_i$$

where n_i is a uniform noise shuch as $\mathbb{P}(n_i = x) = \begin{cases} 0 \text{ if } |x| > \sigma \\ \frac{1}{2\sigma+1} \text{else} \end{cases}$.

The noise depends on one parameter $\sigma \in \mathbb{N}$.

Parameters of the Simulation

Attack

- Key and Textbytes : 8 bits ;
- $\sigma = 24$.

Distinguishers

- Soft Drop Distinguisher : $\gamma = \frac{1}{\hat{q}}$.
- Compare with the Optimal distinguisher (cf. Eq 3).

Results of the Simulation

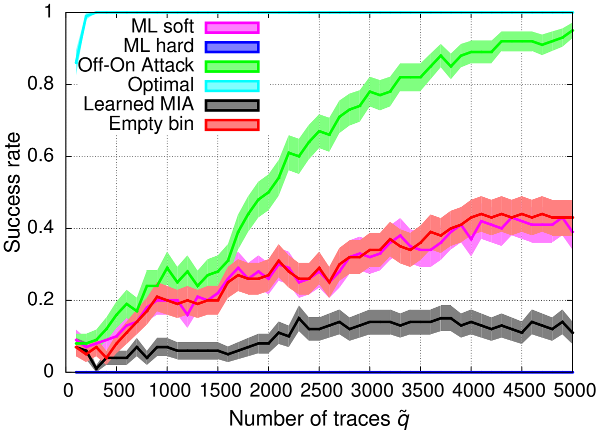


FIGURE: Simulation for $\hat{q} = 320$ and $\sigma = 24$.

Results of the Simulation

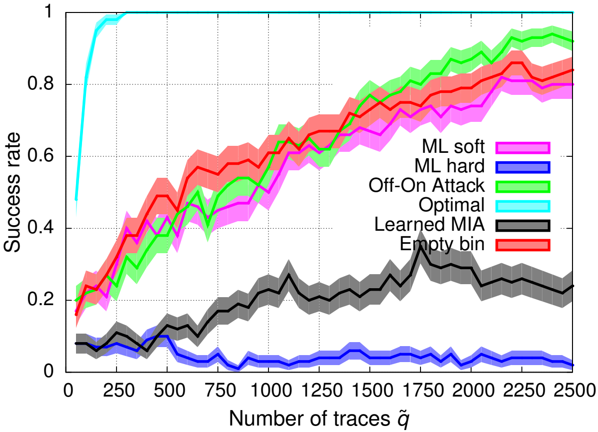


FIGURE: Simulation for $\hat{q} = 320$ and $\sigma = 24$.

Contents

Measurement setup

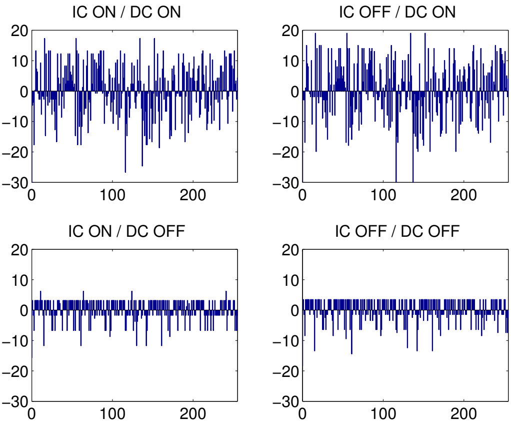
Properties of STM32F407VGT6 microcontroller :

- No *CPU cycle counter* nor *performance register*
- But DWT (Data Watchpoint and Trace) unit has a cycle accurate 32 bit counter (DWT_CYCCNT register)

  $\Longrightarrow$ 10 000 measurements per second.

TELECOM ParisTech

Context : OpenSSL AES *is not* constant time



Apparently, it is not only a matter of caches.

Results on Hardware

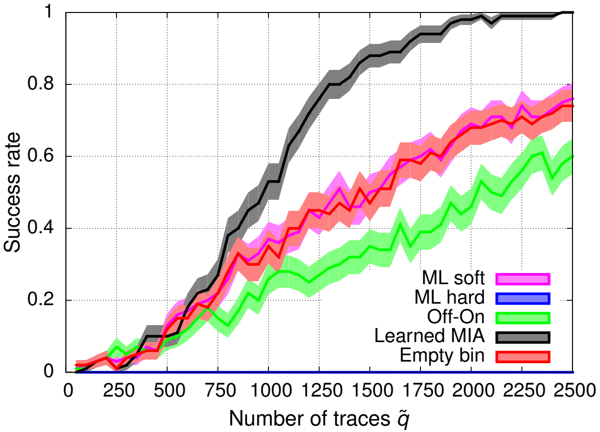


FIGURE: SR for $\hat{q} = 25\,600$ on real-world measurements

Results on Hardware

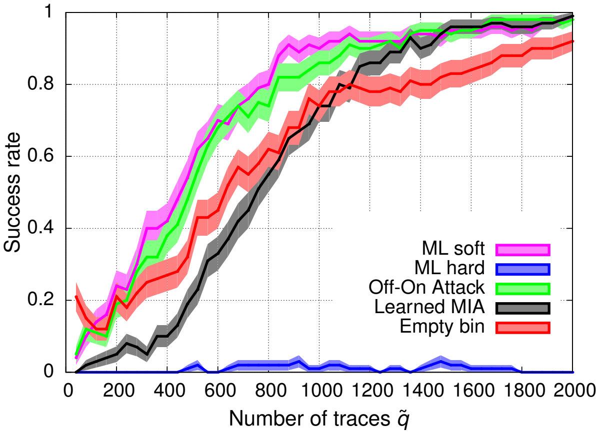


FIGURE: SR for $\hat{q} = 256\,000$ on real-world measurements

Results on Hardware

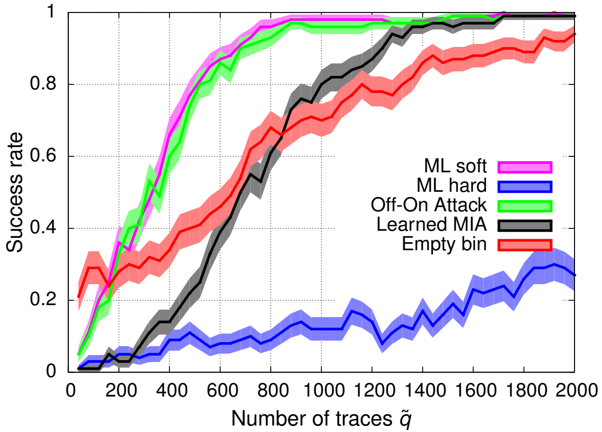


FIGURE: SR for $\hat{q} = 2\ 560\ 000$ on real-world measurements

Contents

Conclusions

- Avoid the Empty Bin issue ;
- Many Distinguishers for Timing attacks ;
- Easy to implement.

TELECOM ParisTech

Thank You

Questions ?

eloi.de-cherisey@mines-telecom.fr

References I

[Ber05] Daniel J. Bernstein.
Cache-timing attacks on AES, April 14 2005.
http://cr.yp.to/antiforgery/cachetiming-20050414.pdf.

[BRM12] Sarani Bhattacharya, Chester Rebeiro, and Debdeep Mukhopadhyay.
Hardware prefetchers leak : A revisit of SVF for cache-timing attacks.
In *45th Annual IEEE/ACM International Symposium on Microarchitecture, MICRO 2012, Workshops Proceedings, Vancouver, BC, Canada, December 1-5, 2012*, pages 17–23. IEEE Computer Society, 2012.

[GBTP08] Benedikt Gierlichs, Lejla Batina, Pim Tuyls, and Bart Preneel.
Mutual information analysis.
In *CHES, 10th International Workshop*, volume 5154 of *Lecture Notes in Computer Science*, pages 426–442. Springer, August 10-13 2008.
Washington, D.C., USA.

References II

[HRG14] Annelie Heuser, Olivier Rioul, and Sylvain Guilley.
Good Is Not Good Enough - Deriving Optimal Distinguishers from Communication Theory.
In Lejla Batina and Matthew Robshaw, editors, *Cryptographic Hardware and Embedded Systems - CHES 2014 - 16th International Workshop, Busan, South Korea, September 23-26, 2014. Proceedings*, volume 8731 of *Lecture Notes in Computer Science*, pages 55–74. Springer, 2014.

[PR09] Emmanuel Prouff and Matthieu Rivain.
Theoretical and Practical Aspects of Mutual Information Based Side Channel Analysis.
In Springer, editor, *ACNS*, volume 5536 of *LNCS*, pages 499–518, June 2-5 2009. Paris-Rocquencourt, France.

[RM12] Chester Rebeiro and Debdeep Mukhopadhyay.
Boosting Profiled Cache Timing Attacks With A Priori Analysis.
Information Forensics and Security, IEEE Transactions on, 7(6) :1900–1905, 2012.

References III

[WHS12] Michael Weiß, Benedikt Heinz, and Frederic Stumpf.
A cache timing attack on AES in virtualization environments.
In Angelos D. Keromytis, editor, *Financial Cryptography and Data Security - 16th International Conference, FC 2012, Kralendijk, Bonaire, Februray 27-March 2, 2012, Revised Selected Papers*, volume 7397 of *Lecture Notes in Computer Science*, pages 314–328. Springer, 2012.