On the Need of Physical Security for Small Embedded Devices: a Case Study with COMP128-1 Implementations in SIM Cards

Yuanyuan Zhou[1], Yu Yu[2], F.-X. Standaert[3], J.-J. Quisquater[3]

[1] Brightsight, Delft, The Netherlands.
[2] East China Normal University and Tsinghua University, China.
[3] ICTEAM/ELEN/Crypto Group, Université catholique de Louvain, Belgium.

Abstract. Ensuring the physical security of small embedded devices is challenging. Such devices have to be produced under strong cost constraints, and generally operate with limited power and energy budget. However, they may also be deployed in applications where physical access is indeed possible for adversaries. In this paper, we consider the case of SIM cards to discuss these issues, and report on successful side-channel attacks against several (old but still deployed) implementations of the COMP128-1 algorithm. Such attacks are able to recover cryptographic keys with limited time and data, by measuring the power consumption of the devices manipulating them, hence allowing cards cloning and communications eavesdropping. This study allows us to put forward the long term issues raised by the deployment of cryptographic implementations. It provides a motivation for improving the physical security of small embedded devices early in their development. We also use it to argue that public standards for cryptographic algorithms and transparent physical security evaluation methodologies are important tools for this purpose.

1 Introduction

Protecting present information systems requires considering both hardware and software security issues, with their specific risks and constraints. In general, software attacks are cheaper and tools for performing them can be rapidly disseminated. Yet, they are also easier to patch with code updates. By contrast, hardware attacks are more difficult to perform, as they require laboratory equipment that ranges from low-cost to highly expensive. But they can be more difficult to fix a posteriori, as hardware updates imply more expensive development processes, and usually take place in the longer term. Hence, finding the best balance between hardware and software security is a difficult task for system designers. This concern is particularly critical with cryptographic implementations that may be the target of fault insertion attacks [2] and side-channel attacks [12, 13, 20]. In the latter case (that will be our focus in this paper), the adversary exploits physical information leakage such as the power consumption of the device running a cryptographic algorithm, in order to extract secret information such as secret keys. As the power consumption of a device is expected to be correlated with the data it manipulates, these attacks essentially proceed by comparing key-dependent leakage predictions with actual measurements. When

no particular care is taken, cryptographic implementations frequently turn out to be highly susceptible to side-channel attacks, as recently exhibited with results against the KeeLoq remote keyless entry systems (at CRYPTO 2009 [8]), the Mifare DESFire contactless smart cards (at CHES 2011 [19]), or Xilinx's FPGA bitstream encryption mechanisms (at ACM CCS 2011 [17]).

Since side-channel attacks do not target algorithms but instances of their implementation in various technologies, it is hard to design general solutions that allow making *any* implementation of an algorithm secure. Hence, state-of-the-art techniques to improve security against such attacks rely on heuristic assumptions (e.g. the masking and hiding in [15]), and need to be confirmed by empirical evaluation. Note that although this situation raises challenging research problems (e.g. discussed at the CHES workshops [6]), producing practically secure integrated circuits is not out of reach. Nowadays, most smart card companies have products evaluated by independent laboratories and granted with high security levels by certification authorities, e.g. [1, 5]. But this improved security usually comes at the cost of implementation overheads that may limit their practical deployment. In addition, and although having certificates may be a good selling point, obtaining them also takes time and money (see, e.g. the Common Criteria [7] and EMVco [9]). Hence, while such certificates are a frequent requirement for security products of government agencies and banking applications, they are less usual in lower-cost applications using SIM or transport cards.

A typical example of this lack of general approaches for preventing side-channel attacks was actually given by a team from IBM in 2002, for implementations of the COMP128-1 algorithm used in GSM communications. In a paper from IEEE S&P [21], Rao et al. first showed that a straightforward application of Differential Power Analysis (DPA) was not successful against the instances of SIM cards they were analyzing (presumably because of some ad hoc countermeasures). Then, they observed that at the first round of COMP128-1's compression function, the substitution-box (S-box) consists of 512 values (i.e. are accessed by a 9-bit index). It implies that on low-speed SIMs (with 8-bit CPU) this S-box has to be implemented using two (typically equal-size) lookup tables. Knowing which table is being accessed (which could be identified from the power traces) could result in a key recovery with a maximum of 1000 random challenges, or 255 chosen ones, or just 8 adaptively chosen ones (i.e. as efficient as a binary search). This data corresponds to the monitoring of a few minutes of SIM card operations. In other words, while the standard DPA approach did not directly lead to successful key recoveries, a slightly modified path taking advantage of the implementation specificities did a perfect job. Fortunately, the attack (exploiting the 8-bit addressing) was only applicable to 8-bit-CPU SIM cards. Since 2003, the major operators have been gradually phasing out the use of legacy SIM by issuing products equipped with 16-bit CPU data bus, ruling out this possibility.

In this paper, we take advantage of this SIM card example to discuss the practical challenges raised by hardware security issues. For this purpose, we investigate the resistance of SIM cards from two different GSM operators and four different manufacturers against DPA. Our experiments target implementations

of the COMP128-1 algorithm in 16-bit CPUs, that are secure against the IBM 2002 attack. They are also secure against the algorithmic collision attacks described in [3]. While COMP128-1 is progressively being replaced by improved versions, it is still deployed in commercial devices, and sometimes being distributed. We show how DPA can be used to recover its 128-bit secret key, allowing cards cloning and communications eavesdropping. Depending on the targets and measurement setup available to the adversary, the attacks require physical access to the device ranging from minutes to a couple of hours. Interestingly, our results can be seen as the methodological counterpart of the 2002 ones. While the previous analysis in [21] targets instances of SIM cards (presumably) secure against standard DPA attacks but weak against dedicated ones, our instances are robust against the IBM attack but weak against standard DPA.

The important conclusions of this work are methodological. First, our results exhibit the long term nature of physical security concerns. While cryptographic implementations are not deployed as long as algorithms, they may remain in service for a couple of years, and are not straightforward to upgrade. This observation makes a case for considering physical security as an important feature of small embedded devices in general. Technical solutions exist to make side-channel attacks significantly more difficult to perform, e.g. the previously mentioned masking and hiding. But they work best if considered early in a design process. Second, we observe that public standards for cryptographic algorithms are useful to improve the efficiency of countermeasures against physical attacks. By contrast, the closed-source nature of COMP128-1 has significantly limited the amount of research about its secure implementations. Finally, transparent and reproducible (possibly standardized) methodologies for physical security evaluations are required, in order to quantify physical security on a sound basis.

Contact with the operators. Our experiments have been performed in 2010. The different operators exploiting the SIM cards that we discuss in this paper have been contacted before publication of our results. Updates towards implementations of COMP128-2 and COMP128-3, including protections against side-channel attacks, are under development (or maybe already deployed).

2 Background

For place constraints, details about the GSM infrastructure and previous works on SIM cloning fraud and countermeasures have been deferred to the long version of the paper [25]. In this section, we briefly recall the processing of the compression function in COMP128-1 and necessary basics on side-channel attacks.

COMP128-1 is a cryptographic hash function that takes a 32-byte input (i.e. a 16-byte challenge RAND and a 16-byte secret key KI), and produces a 12-byte output by iterating 8 rounds. In our attacks, the most important part of this algorithm is its compression function that consists of 5 (sub-)rounds that combine the key material and randomness. In particular, the sensitive operations that we will target are the following data update occurring in the first round:

$$y = (\mathsf{KI}[m] + 2{\cdot}\mathsf{RAND}[m]) \bmod 2^{9-j},$$
$$z = (2{\cdot}\mathsf{KI}[m] + \mathsf{RAND}[m]) \bmod 2^{9-j},$$

that occur for $0 \leq m \leq 15$ secret key bytes (with j the RAND byte index).

Side-channel attacks generally exploit the existence of data-dependent and physically observable phenomenons caused by the execution of computing tasks in present microelectronic devices. Typical examples of such information leakages include the power consumption and the electromagnetic radiation of integrated circuits. We will focus on the first one in the rest of this paper. The literature usually divides such attacks in two classes. First, Simple Power Analysis (SPA) attempts to interpret the power consumption of a device and deduce information about its performed operations. This can be done by visual inspection of the power consumption measurements in function of the time. SPA in itself does not always lead to key recovery. For example with block ciphers, distinguishing the encryption rounds does not reveal any sensitive information. Yet, it is usually an important preliminary step in order to reduce the computational requirements of more advanced attacks. Second, Differential Power Analysis (DPA) intends to take advantage of data-dependencies in the power consumption patterns. In its standard form [16], DPA is based on a divide-and-conquer strategy, in which the different parts of a secret key (usually denoted as "subkeys") are recovered separately. The attack is best illustrated with an example. Say one targets the first round of a block cipher, where the plaintext is XORed with a subkey and sent through a substitution box S. DPA is made of three main steps:

1. For different plaintexts x_i and subkey candidates k^*, the adversary predicts intermediate values in the target implementation. For example, one could predict S-box outputs and get values $v_i^{k^*} = \mathsf{S}(x_i \oplus k^*)$.
2. For each of these predicted values, the adversary models the leakages. For example, if the target block cipher is implemented in a CMOS-based microcontroller, the model can be the Hamming weight (HW) of the predicted values. One then obtains modeled leakages $m_i^{k^*} = \mathsf{HW}(v_i^{k^*})$.
3. For each subkey candidate k^*, the adversary compares the modeled leakages with actual measurements, produced with the same plaintexts x_i and a secret subkey k. In the univariate DPA attacks (that we will apply), each $m_i^{k^*}$ is compared independently with many single points in the traces, and the subkey candidate that performs best is selected by the adversary.

Different statistical tools have been proposed to perform this comparison. In our experiments, we will consider a usual DPA distinguisher, namely Pearson's correlation coefficient [4]. In this case, and denoting a leakage sample produced with plaintext x_i and subkey k as l_i^k, the adversary selects the subkey candidate as:

$$\tilde{k} = \underset{k^*}{\operatorname{argmax}} \frac{\sum_i (m_i^{k^*} - \overline{m}^{k^*}) \cdot (l_i^k - \overline{l}^k)}{\sqrt{\sum_i (m_i^{k^*} - \overline{m}^{k^*})^2 \cdot \sum_i (l_i^k - \overline{l}^k)^2}},$$

where $\overline{m}^{k^*}$ and $\overline{l}^k$ are the sample means of the models and leakage samples. The complete master key is recovered by repeating this procedure for every subkey.

3 DPA attacks against implementations of the COMP128-1 algorithm in SIM cards

3.1 Target SIM cards & measurement setup

In this section, we perform DPA attacks on four representative SIM cards denoted as #1,#2, #3 and #4. Besides corresponding to various operators and manufacturers, the main difference between these implementations is that they sometimes include protections against the algorithmic collision attacks described in [3], next denoted as the "Indexed Challenges" (I-C) and "Collision Free" (C-F) countermeasures. The details of these countermeasures are not necessary for the understanding of the paper, but are given in the long version [25]. As summarized in Table 1, SIM#1 and SIM#2 are susceptible to collision attacks in 20 000 and more queries, SIM#3 and SIM#4 are immune against them.

Table 1. Target SIM cards.

	Manufact.	Operator	Countermeasure(s)
SIM#1	I	A	Not Available
SIM#2	II	B	I-C
SIM#3	III	B	I-C + C-F
SIM#4	IV	B	I-C + C-F

We used a LeCroy WavePro 950 oscilloscope to acquire the power traces, via a small resistor of 25 Ohm between the GND of power supply and the GND of a self-made Card-to-Terminal adapter. The Card-to-Terminal adapter was tweaked to provide an external DC power to the test card via a Kenwood P18A power supply (+5V), and to provide an external clock to the card via an Agilent 33120A function generator(5MHz Frequency, 2.2V Amplitude and 1.1V Offset). We used a commercially available card reader and software to control the test card during the acquisitions. In addition, we used a Keithley 488 GPIB card (i.e. a PCI card installed inside a PC) to communicate with the oscilloscope.

3.2 Preprocessing of the traces

As usual when implementing side-channel attacks, we started by applying SPA in order to identify the relevant parts of the power traces. This task is easy for SIM#1 and SIM#2. As shown in the left part of Figure 1, we can identify the 8 iterative rounds of COMP128-1 by visual inspection. By further zooming on the different iterations, we could even observe the 5 sub-rounds of the COMP128-1 compression function (see Figure 2 in [25]). Therefore, it is directly possible to extract the parts of the power traces where to apply DPA for these two targets. The situation slightly differs for SIM cards #3 and #4, where the Collision Free countermeasure was implemented. As illustrated in the right part of Figure 1 (and Figure 6 in [25]), it is again possible to identify the COMP128-1 operations

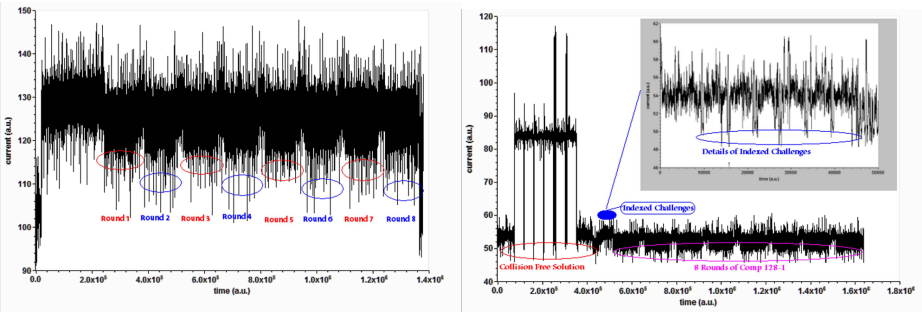

Fig. 1. Left: a power trace from SIM#1. Right: a power trace from SIM#3.

(as well as the Indexed Challenges) in the power traces. Yet, the Collision Free countermeasure includes a randomized memory writing operation (i.e. it uses randomness to decide whether to store a current request or not). Therefore, the length of the power traces varies for different inputs, which requires special care for aligning the traces after acquisition. In order to deal with this situation, a simple solution is to apply pattern matching techniques. That is, we selected a characteristic pattern including the samples of interest for our DPA attacks, and then systematically identified them in following traces using cross-correlation. As the noise level in our measurements was relatively low, such a simple heuristic was sufficient for performing successful key recoveries, as will be described next.

3.3 DPA attack results

Since no countermeasures in our target SIM cards prohibit random queries, we generated our traces by repeatedly executing the COMP128-1 algorithm with such inputs. Next, we applied exactly the divide-and conquer strategy focusing on the intermediate values y and z at the first sub-round of the first round in the implementation of COMP128-1, as described in Section 2. For each $0 \leq m \leq 15$, we built predictions for the 256 possible values of $\mathsf{KI}[m]$ and performed the comparison. The result of such a comparison for one of the 16 COMP128-1 subkeys is given in Figure 2 for SIM#2 and SIM#3 (similar results are given for the other targets in [25]). The figures plot the value of Pearson's correlation coeffi-

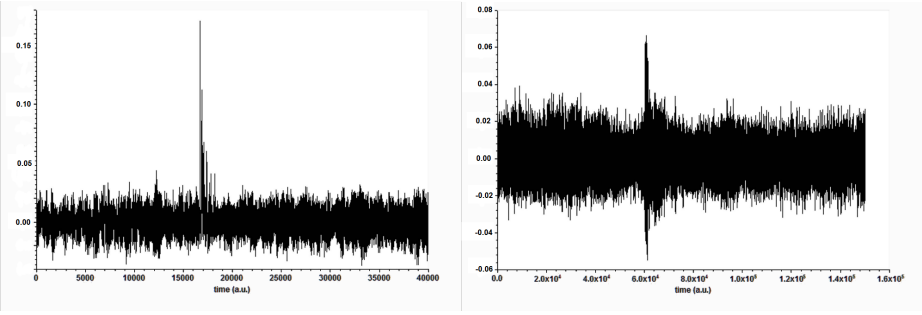

Fig. 2. Left: DPA result against SIM#2. Right: DPA result against SIM#3.

cient over time, using y as a target value. We observe that a significant peak is distinguishable at the time samples where the computation of y actually takes place, and this peak only appears for the correct subkey candidate. As expected, the situation was slightly more challenging for SIM#3 and SIM#4. This is due to more noisy traces and the previously mentioned synchronization issue. Yet, in both cases, a DPA peak remained clearly distinguishable, and we could always identify the COMP128-1 subkeys. Finally, we consistently recovered the full key of SIM#1 and SIM#2 with an amount of traces in the hundreds range, and this number extends to the thousands range for SIM#3 and SIM#4. These estimated data complexities are in accordance with the work of Mangard at CT-RSA 2004 [14], where it is shown that the number of measurement traces needed to recover a subkey is inversely proportional to the square of the correlation coefficient estimated for the correct key candidate. In practice, these data complexities corresponds to a few minutes to a couple of hours of acquisition.

4 Conclusions & future work

Technically, it is not a surprise that weakly protected chips can be defeated by side-channel attacks. Yet, our results exhibit (or recall) that such attacks are relatively easy to implement, and are certainly accessible to determined adversaries. Taking the example of SIM cards, this can have severe consequences for the security of GSM communications. Overall, the security of a system is always as strong as its weakest point. Hence, distributing cryptographically-enhanced chips without a sufficient care for physical security leads to unbalanced situations, as side-channel attacks may constitute a trapdoor to circumvent mathematical security. This is especially important for small embedded devices, for which physical access may sometimes be granted to adversaries. In this respect, it is more surprising that (somewhat) security sensitive applications do not always build on certified chips (following what is done, e.g. for bank cards). Admittedly, the target SIM cards investigated in this paper implement old versions of the GSM algorithms, in old technologies. Nevertheless, some of these cards are still in circulation and cards cloning is an important concern that could prevent the adoption of new services. Hence, this situation illustrates the long term nature of hardware security issues. It provides a general motivation for considering them as an important element to take into account early in cryptographic developments. In this respect, we note that the use of proprietary algorithms in commercial products significantly slows down progresses in securing their implementation. In view of the implementation-specific nature of physical attacks, it frequently turns out that protection mechanisms that are tailored to certain cryptographic algorithms provide the best efficiency vs. security tradeoffs. For example, secure implementations of the AES have been the subject of a large literature over the last 10 years. By contrast, no similar analysis is available for COMP128-1. Worse, the use of large (e.g. 512-bit) tables makes it hardly suitable for implementation of countermeasures such as software masking [10]. Following this observation and in the long term, considering protections against physical attacks as a design criteria for cryptographic algorithms could be useful.

While resorting to certification would be an important step in improving the security of SIM cards, we finally note that the procedures used by evaluation laboratories could also benefit from an improved transparency. That is, currently certified chips certainly rule out the possibility of simple attacks as we describe in this paper. But it remains that the exact security level they guarantee is opaque for the end-users, and this opaqueness generally increases as countermeasures are added to the chips. Proposals of worst-case security evaluations aiming at limiting the risks of a "false sense" of security could improve this situation [22, 23]. Considering the strongest available adversaries and taking advantage of the latest cryptanalytic progresses during evaluations of cryptographic hardware appears important in view of the difficulty to fix physical security breaches a posteriori. Eventually, better reflecting side-channel evaluation methodologies in public standards would be highly beneficial too. In this respect, it is noticeable that the ISO 19790 draft standard on "security requirements for cryptographic modules" (aka. FIPS-140-3 [18]) leaves the section on non-invasive attack methods essentially optional to vendors, with little details about the evaluation procedures.

Acknowledgements. Yu Yu was supported by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301, the National Natural Science Foundation of China Grant 61033001, 61172085, 61061130540, 61073174, 61103221, 11061130539, 61021004 and 61133014. F.-X. Standaert is an associate researcher of the Belgian Fund for Scientific Research (FNRS-F.R.S.). This work has been funded in part by the ERC project 280141 on CRyptographic Algorithms and Secure Hardware.

References

1. ANSSI. Agence nationale de la securite des systemes d'information, http://www.ssi.gouv.fr/en/products/certified-products/, retrieved on feb. 1, 2012.
2. Dan Boneh, Richard A. DeMillo, and Richard J. Lipton. On the importance of checking cryptographic protocols for faults (extended abstract). In Walter Fumy, editor, *EUROCRYPT*, volume 1233 of *LNCS*, pages 37–51. Springer, 1997.
3. Mark Briceno, Ian Goldberg, and David Wagner. GSM Cloning. `http://www.isaac.cs.berkeley.edu/isaac/gsm-faq.html`, 1998. Retrieved on Oct. 14, 2011.
4. Eric Brier, Christophe Clavier, and Francis Olivier. Correlation power analysis with a leakage model. In Marc Joye and Jean-Jacques Quisquater, editors, *CHES*, volume 3156 of *Lecture Notes in Computer Science*, pages 16–29. Springer, 2004.
5. BSI. Federal office for information security, https://www.bsi.bund.de/en/topics/certification/certification_node.html, retrieved on feb. 1, 2012.
6. CHES. http://www.chesworkshop.org/.
7. Common Criteria. http://www.commoncriteriaportal.org/.
8. Thomas Eisenbarth, Timo Kasper, Amir Moradi, Christof Paar, Mahmoud Salmasizadeh, and Mohammad T. Manzuri Shalmani. On the power of power analysis in the real world: A complete break of the keeloqcode hopping scheme. In David Wagner, editor, *CRYPTO*, volume 5157 of *LNCS*, pages 203–220. Springer, 2008.
9. EMVco. http://www.emvco.com/. Retrieved on April 11, 2012.
10. Louis Goubin and Jacques Patarin. Des and differential power analysis (the "duplication" method). In Çetin Kaya Koç and Christof Paar, editors, *CHES*, volume 1717 of *Lecture Notes in Computer Science*, pages 158–172. Springer, 1999.

11. Antoine Joux, editor. *Advances in Cryptology - EUROCRYPT 2009, 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, April 26-30, 2009. Proceedings*, volume 5479 of *Lecture Notes in Computer Science*. Springer, 2009.
12. Paul Kocher. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In Neal Koblitz, editor, *Advances in Cryptology—CRYPTO '96*, volume 1109 of *LNCS*, pages 104–113. Springer-Verlag, 18–22 August 1996.
13. Paul Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. In Wiener [24], pages 388–397.
14. Stefan Mangard. Hardware countermeasures against dpa ? a statistical analysis of their effectiveness. In Tatsuaki Okamoto, editor, *CT-RSA*, volume 2964 of *Lecture Notes in Computer Science*, pages 222–235. Springer, 2004.
15. Stefan Mangard, Elisabeth Oswald, and Thomas Popp. *Power analysis attacks - revealing the secrets of smart cards*. Springer, 2007.
16. Stefan Mangard, Elisabeth Oswald, and Francois-Xavier Standaert. One for all – all for one: unifying standard differential power analysis attacks. *IET Information Security*, 5(2):100–110, 2011.
17. Amir Moradi, Alessandro Barenghi, Timo Kasper, and Christof Paar. On the vulnerability of fpga bitstream encryption against power analysis attacks: extracting keys from xilinx virtex-ii fpgas. In Yan Chen, George Danezis, and Vitaly Shmatikov, editors, *ACM CCS*, pages 111–124. ACM, 2011.
18. National Institute of Standards and Technologies. http://csrc.nist.gov/publications/PubsDrafts.html. Retrieved on March 25, 2012.
19. David Oswald and Christof Paar. Breaking mifare desfire mf3icd40: Power analysis and templates in the real world. In Bart Preneel and Tsuyoshi Takagi, editors, *CHES*, volume 6917 of *LNCS*, pages 207–222. Springer, 2011.
20. Jean-Jacques Quisquater and David Samyde. Electromagnetic analysis (EMA): Measures and counter-measures for smart cards. In *Smart Card Programming and Security (E-smart 2001) Cannes, France*, volume 2140 of *LNCS*, pages 200–210, September 2001.
21. Josyula R. Rao, Pankaj Rohatgi, Helmut Scherzer, and Stephane Tinguely. Partitioning attacks: Or how to rapidly clone some gsm cards. In *IEEE Symposium on Security and Privacy*, pages 31–44, 2002.
22. François-Xavier Standaert. Some hints on the evaluation metrics & tools for side-channel attacks. proceedings of the nist non-invasive attacks testing workshop, nara, japan, september 2011.
23. François-Xavier Standaert, Tal Malkin, and Moti Yung. A unified framework for the analysis of side-channel key recovery attacks. In Joux [11], pages 443–461.
24. Michael Wiener, editor. *Advances in Cryptology—CRYPTO '99*, volume 1666 of *LNCS*. Springer-Verlag, 15–19 August 1999.
25. Yuanyuan Zhou, Yu Yu, Francois-Xavier Standaert, and Jean-Jacques Quisquater. On the Need of Physical Security for Small Embedded Devices: a Case Study with COMP128-1 Implementations in SIM Cards (long version).