Formal Study of a Countermeasure Against Side-Channel Attacks

Location. SEN group of the COMELEC department at Telecom ParisTech (37 rue Dareau, 75014, Paris). This internship can lead to a PhD thesis.

Advisors. Pablo Rauzy and Sylvain Guilley ({*firstname.lastname*}@telecom-paristech.fr).

Context. In applications such as cryptography or real-time systems, formal methods are used to prove functional properties on the critical parts of the code. Specifically in cryptography, some non-functional properties are also important, but are not typically certified by formal proofs yet. One example of such a property is the resistance to side-channel attacks. Side-channel attacks are a real world threat to cryptosystems; they exploit auxiliary information gathered from implementations through physical channels such as power consumption, electromagnetic radiations, or time, in order to extract sensitive information (e.g., secret keys) [3, 2, 4].

Many existing countermeasures against side-channel attacks are implemented at the hardware level, especially for smartcards. However, software level countermeasures are also very important, not only in embedded systems where the hardware cannot always be modified or updated, but also in the purely software world [9, 6]. This motivates the search for formal methods that can certify the resistance of software to side-channel attacks.

Most existing countermeasures rely heavily on randomness (e.g., masking [5, Chp. 9]) which is costly, especially on embedded systems such as smartcards, where resources are scarce. Nonetheless it has recently been shown that it is possible and practical to build countermeasures that do not require any randomness [1, 8] (hiding or balancing [5, Chp. 7]), and to prove their security, including in software. Indeed, we were able to significantly decrease the signal-to-noise ratio of the side-channel analysis attacker using a technique known as *dual-rail with precharge logic* (DPL) [7]. The idea of DPL is to compute on redundant representations using two indistinguishable resources (noted y_{true} and y_{false}), so that the attacker cannot know which one has been set. This approach works well but not perfectly since actual hardware leakages can be more subtle than what the leakage model can express. However, as the perspectives mentioned in [7] explain, adding randomness would help to significantly improve the relevance of the leakage model.

Problems.

— Developing the new formal proofs that are needed when adding randomness. Indeed, the goal is no longer to avoid leakage altogether but rather to have indistinguishable distributions.

— Studying the feasibility of masking on top of the DPL has to be studied (cost, proof, etc.) for different (combination of) approaches: randomly swapping y_{true} and y_{false}, randomly choosing the pair of bits used for the DPL protocol, randomly padding the unused bits, using an existing masking scheme on top of the DPL protocol.

— Porting and validating in practice the DPL countermeasures onto other targets: other cryptographic algorithms and other hardware platforms, such as AES on ARM for instance.

Organization. During this internship the candidate should:

1. Familiarize herself/himself with:
   - the literature on side-channel attacks and especially power analysis;
   - the paioli[1] framework;
   - masking countermeasures.
2. Formally express and prove (discrete) probabilistic properties on symbolic execution.
3. Explore the different problems we listed and implement at least one of the perspectives.
4. Write a paper and submit it to a peer-reviewed conference ☺.

[1] `http://pablo.rauzy.name/sensi/paioli.html`

References

[1] Cong Chen, Thomas Eisenbarth, Aria Shahverdi, and Xin Ye. Balanced Encoding to Mitigate Power Analysis: A Case Study. In *CARDIS*, Lecture Notes in Computer Science. Springer, November 2014. Paris, France.

[2] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. In *Proceedings of CRYPTO'96*, volume 1109 of *LNCS*, pages 104–113. Springer-Verlag, 1996.

[3] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential Power Analysis. In *Proceedings of CRYPTO'99*, volume 1666 of *LNCS*, pages 388–397. Springer-Verlag, 1999.

[4] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. In Michael J. Wiener, editor, *CRYPTO*, volume 1666 of *Lecture Notes in Computer Science*, pages 388–397. Springer, 1999.

[5] Stefan Mangard, Elisabeth Oswald, and Thomas Popp. *Power Analysis Attacks: Revealing the Secrets of Smart Cards*. Springer, December 2006. ISBN 0-387-30857-1, `http://www.dpabook.org/`.

[6] Luke Mather and Elisabeth Oswald. Pinpointing side-channel information leaks in web applications. *J. Cryptographic Engineering*, 2(3):161–177, 2012.

[7] Pablo Rauzy, Sylvain Guilley, and Zakaria Najm. Formally Proved Security of Assembly Code Against Power Analysis. Cryptology ePrint Archive, Report 2013/554, 2013. `http://eprint.iacr.org/2013/554`.

[8] Victor Servant, Nicolas Debande, Houssem Maghrebi, and Julien Bringer. Study of a Novel Software Constant Weight Implementation. In *CARDIS*, Lecture Notes in Computer Science. Springer, November 2014. Paris, France.

[9] Yinqian Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. Cross-VM side channels and their use to extract private keys. In Ting Yu, George Danezis, and Virgil D. Gligor, editors, *ACM Conference on Computer and Communications Security*, pages 305–316. ACM, 2012.