Leakage Assessment on Ascon_first_order

10/17/2022

1. Target

a) Algorithm: Ascon
b) Implementer: **Ruhr-University Bochum, Germany**
c) Variant: Ascon128v12_first_order
d) URL: **https://github.com/Chair-for-Security-Engineering/LWC-Masking**
e) Commit hash: **4244b255e282e2d309aa270960bcd5a594c2db03**
f) Protection method: **Hardware Private Circuits 2 (HPC2)**
g) Protection order: **1**

2. Equipment and Software Used

(a) General type of the evaluation platform: **Control board is FOBOS3 board for control. This board uses a PYNQ-Z1 board with a Zynq SoC (XC7Z020-1CLG400C) and a custom board that hosts an ADC for power measurement. The target board is NewAE CW305 Aritx7 (xc7a100tftg256)**
(b) Oscilloscope and its major characteristics: **We utilized OpenADC with 40 MHz bandwidth and a maximum sampling rate of 100 MS.**
(c) Current and electromagnetic probes: **N/A**
(d) Usage of bandwidth limiters, filters, amplifiers, etc. and their specification: **Power was measured at the output of the CW305's onboard amplifier which amplifies the voltage drop across the board's shunt resistor.**
(e) Are sampling clock and design-under-evaluation clock synchronized? **Yes**
(f) Names and versions of programs used for evaluating side-channel resistance: **FOBOS3 analysis software.**
g) Clock frequency of target: **16 MHz**
h) Sampling frequency and resolution: **80M Sample/sec sampling frequency and 10 bit resolution.**

3. Leakage Assessment Method

a) Type of the method: **Fixed-vs-random Test Vector Leakage Assessment [GJJR11 ,SM15].**
b) Number of traces: **10 million traces.**
c) Source of randomness: **Trivium-based DRBG.**
d) Trigger location relative to the execution start time of the algorithm: **Triggered ADC at the start of the algorithm.**
e) Time required to collect data for a given attack/leakage assessment: ~27 Hrs**.**
f) Total time of the attack/assessment: ~27 Hrs.
g) Total size of all traces (if stored): **Not stored.**
h) Availability of raw measurement results: **N/A.**
i) Test vector generation:
The test vectors used and the scripts used for generation are available in the attached file.

The procedure is as follows:

1- Generate a short unshared test vector using cryptotvgen.

2- Convert it into a shared format using gen_shared.py.

3- Generate the FOBOS-ready fixed-vs-random test vectors using lwc_2_fobos_tv.py. The SDI is kept similar in all test vectors, and the PDI section is fixed in fixed test vectors and random in the random test vectors. Fresh sharing on the PDI section is generated in all test vectors.

4. Results:

a) Documentation of results: **Figure 1 shows first-order TVLA using 10 million traces. Figure 2 shows t-values vs. number of traces processed. Listing 1 shows the clock cycles where t-values exceed the threshold for the test vectors attached.**

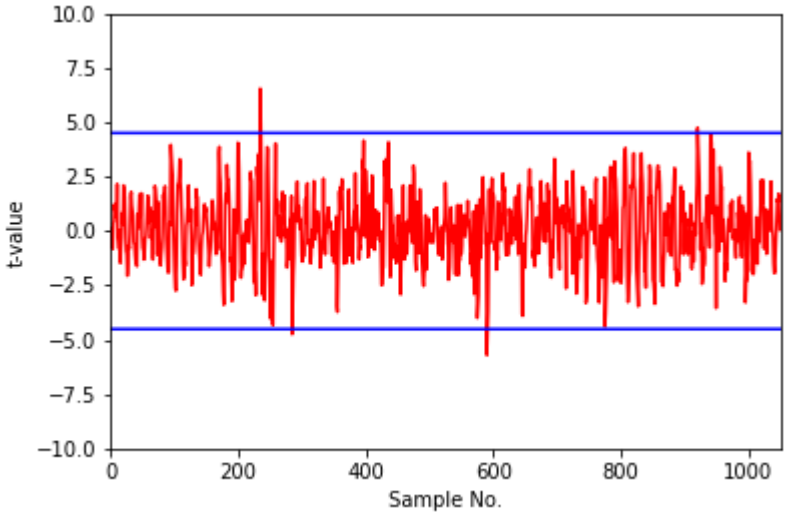

Figure 1 - Ascon_first_order TVLA (10 million traces)

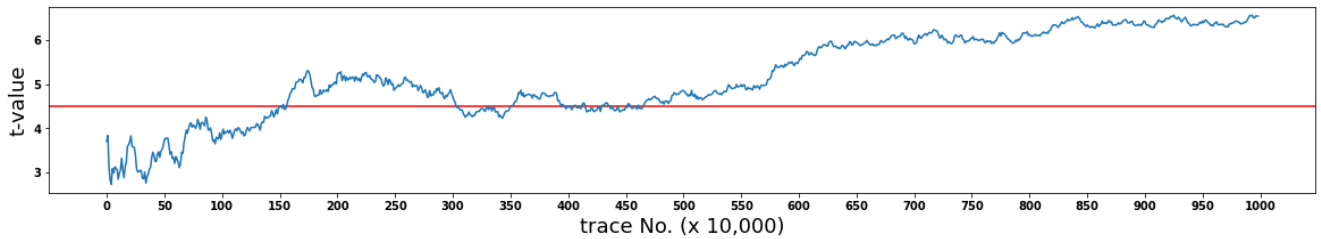

Figure 2 - Ascon_first_order maximum t-value vs. number of traces (x 10,000)

Listing 1 : Samples that exceed the 4.5 threshold
sample - - clk - - tvalue
235 --47 -- 6.5
236 --47 -- 5.2
285 --57 -- -4.7
590 --118 -- -5.7
591 --118 -- -4.6
920 --184 -- 4.7

References

[GJJR11] G. Goodwill, B. Jun, J. Jaffe, and P. Rohatgi, “A testing methodology for side-channel resistance validation,” Nara, Japan, 2011.

[SM15] T. Schneider and A. Moradi, “Leakage Assessment Methodology - a clear roadmap for side-channel evaluations,” Cryptology ePrint Archive 2015/207, Jun. 2015. Accessed: Dec. 31, 2021. [Online]. Available: https://eprint.iacr.org/2015/207