PB173 Domain specific development: side-channel analysis

Seminar 6: Progress on First Steps

Łukasz Chmielewski
chmiel@fi.muni.cz,

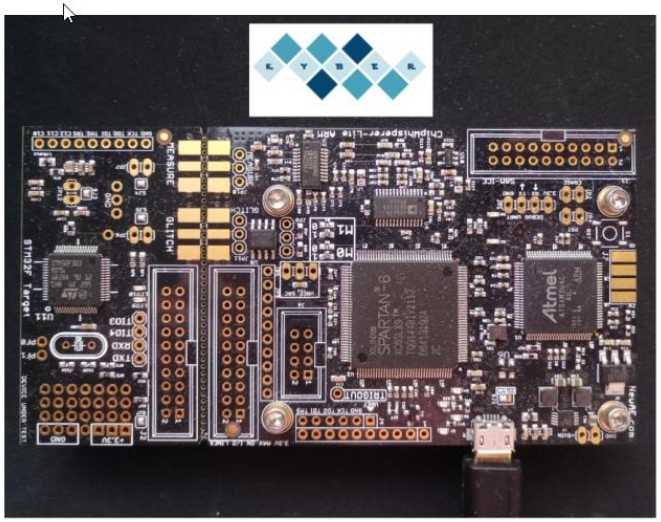
Consultation: A406 Friday 9:00-11:00

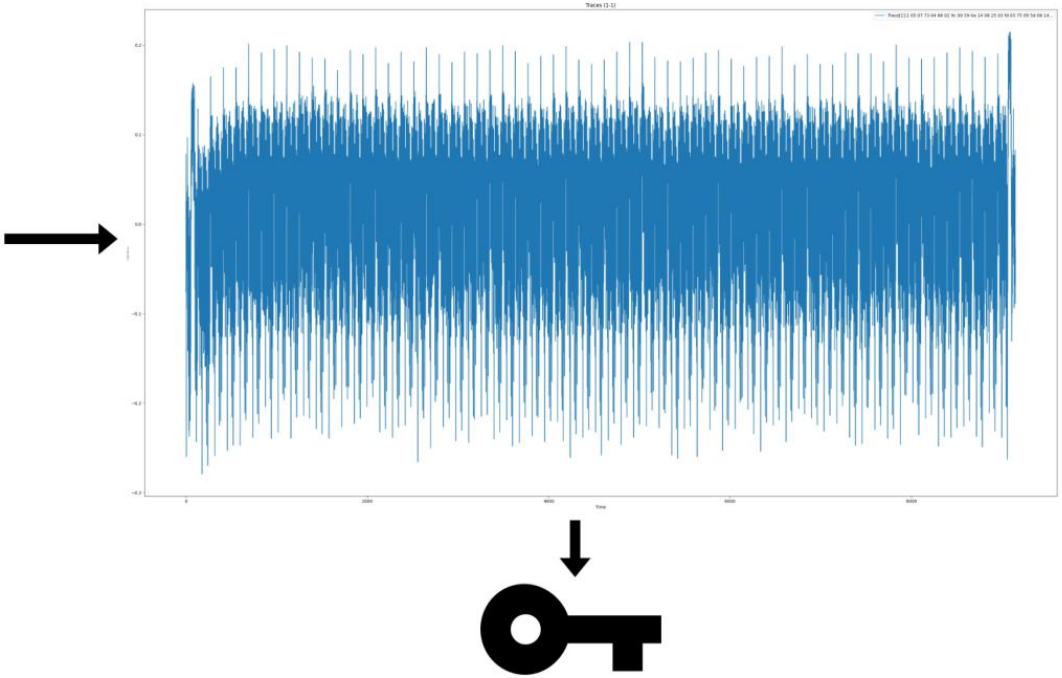
CROCS
Centre for Research on Cryptography and Security

EXAMPLES

1: Attacks on Kyber

- Post-quantum key encapsulation mechanism

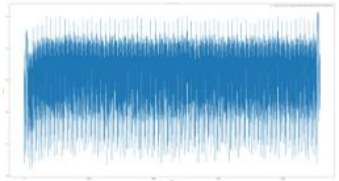




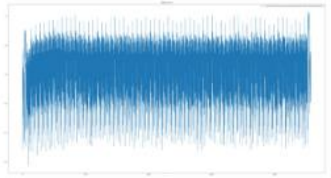
1: Attacks on Kyber

- Many multiplications based on 2 bytes values (to be precise 12 bits)
- Template Matching attack: “Breaking DPA-protected Kyber via the pair-pointwise multiplication”, recently submitted, see https://eprint.iacr.org/2023/551

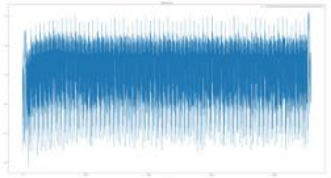
$$\underset{\text{Mult}_0}{0 * 0} + \underset{\text{Mult}_1}{0 * 0} + \underset{\text{Mult}_2}{0 * 0} + \cdots + \underset{\text{Mult}_{127}}{0 * 0}$$



$$\underset{\text{Mult}_0}{0 * 1} + \underset{\text{Mult}_1}{0 * 1} + \underset{\text{Mult}_2}{0 * 1} + \cdots + \underset{\text{Mult}_{127}}{0 * 1}$$



$$\underset{\text{Mult}_0}{3328 * 3328} + \underset{\text{Mult}_1}{3328 * 3328} + \underset{\text{Mult}_2}{3328 * 3328} + \cdots + \underset{\text{Mult}_{127}}{3328 * 3328}$$



1: Attacks on Kyber

- Let’s open:
  - fixedTraces-2023-06-14-01-53-01+SAVED3.zip
- Visualize and perform correlation.
- What is wrong with the trace?
- What would have to be done to make it better?

2: PIN Checking simple_pin.c: find two problems

```
1  char realPukPin[] = { ... };
2  short counter;//variable to store the current counter value; it is being read and stored from / to flash
3
4  bool checkPin(char[] pin, short offset, short length) {
5      if (cardState == BLOCKED)
6          return false;
7
8      readCounterFromFlash(&counter);//read counter value from flash
9      //realPukPin+PUK_LENGTH points to the PIN
10     if ((counter > 0) && (! memcmp(pin+offset, realPin, length)))
11     {
12         counter = counterLimit;
13         writeCounterToFlash(counter);//program counter value to flash
14         return true;
15     }
16     counter--;
17     writeCounterToFlash(counter);
18     return false;
19 }
20
21 void memcpy(void *dest, void *src, size_t n)
22 {
23     // Typecast src and dest addresses to (char *)
24     char *csrc = (char *)src;
25     char *cdest = (char *)dest;
26
27     // Copy contents of src[] to dest[]
28     for (int i=0; i<n; i++)
29         cdest[i] = csrc[i];
30 }
```

ORGANIZATIONAL

Division

- Group 1: Tomas Re, Tomas Ro, Martin
  - Topic: Visualization
  - GitHub repository (quite empty): https://github.com/reznakt/pb173-sca-visualization
- Group 2: Michael T, Lubomir, Richard
  - Topic: Standard Processing, Michael might touch also “Parallel computations with acquisition”
  - GitHub repository (some work seems to be there): https://github.com/LubJur/PB173_standard_signal_processing
- Group 3: Michal, Matus, Filip
  - Topic: Align
  - GitHub repository (just started): https://github.com/mr-akiio/trs-alignment

Please register in IS

Lukasz Michal Chmielewski, PhD	
1.	Align
☐	*Supervisor:* Lukasz Michal Chmielewski, PhD, učo 247858 *Students (max. 3)*: 1. Michal Bahna, učo 536283, **FI** B-PVA PVA [sem 3, year 2] 2. Matúš Renko, učo 536653, **FI** B-PVA PVA [sem 3, year 2]
	Display operations
2.	Signal Processing
☐	*Supervisor:* Lukasz Michal Chmielewski, PhD, učo 247858 *Students (max. 4)*: 1. Ľubomír Jurčišin, učo 536638, **FI** B-INF IN [sem 3, year 2] 2. Michael Trávníček, učo 535360, **FI** B-INF IN [sem 3, year 2]
	Display operations
3.	Visualization
☐	*Supervisor:* Lukasz Michal Chmielewski, PhD, učo 247858 *Students (max. 3)*: 1. Martin Lubojacký, učo 524912, **FI** B-PVA PVA [sem 5, year 3] 2. Tomáš Režňák, učo 525055, **FI** B-PVA PVA [sem 5, year 3] 3. Tomáš Rohlínek, učo 524880, **FI** B-PVA PVA [sem 5, year 3]
	Display operations

Reminder: Colloquium

- To get the colloquium
  - You must be present at seminars (2 absences OK)
  - You must be active at seminars (+2 points given by me at the end)
  - You must submit and get:
    - 50%: 7 points in total

      (projects + presentation + activity = 14 points)

Seminars Plan

- 7: today, no points
- 8: evaluation of first steps given last week: 3 points per group (personalized per person based on the Github)
  + Giving new tasks
- 9: -
- 10: 4 points per group (personalized per person based on the GitHub)
  + Giving new tasks
- 11: presentations about work done + work in progress
- 12: final 2 points for work
  + 2 points for activity, grading.

WORK

Group 1: Standard Signal Processing

- First Tasks:
  - Implement easy modules: average, standard deviation, histogram, absolute value,
  - You can have a look at SaveAs.py and correlation.py
  - Try to implement computing spectrum, some inspiration: https://realpython.com/python-scipy-fft/
- How is it going?

Group 2: Visulation

- First Tasks: implement displaying traces using 2-3 different libraries
  - Matplotlib, bokeh, search for more
  - Someone did some work on that. Have a look here, but it might be chaotic: https://github.com/nilswiersma/pywf/tree/master
- How is it going?

Group 3: Alignment

- First tasks:
  - investigate cross-correlations in python
  - See all the uploaded scripts
  - Especially SaveAs.py and correlation.py
- How is it going?
- If you have issues then we can discuss peak-based alignment (that can be put as first task too).

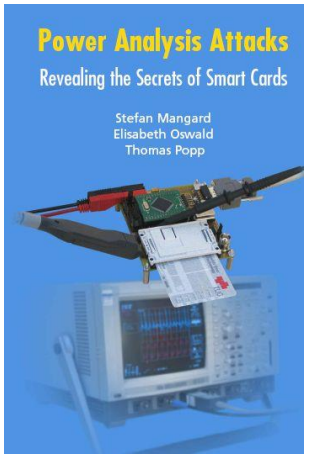
Parallel computations with acquisition (?)

- First Task: measure the efficiency of the acquisition
- Should we discuss it?

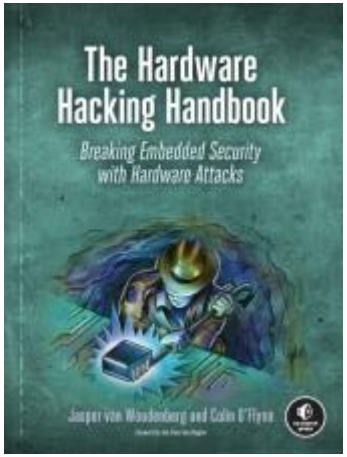
WORK IN GROUPS (60 MIN)

Reading

- For interested people
- Side-Channel Analysis – blue book:
  - http://dpabook.iaik.tugraz.at/
  - The books is available at the uni.
  - Look online



- The Hardware Hacking Handbook:
  - https://nostarch.com/hardwarehacking
  - I have an epub version.



Questions