INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES TOULOUSE

01/06/2023

Research project presentation

Lost In Time

Timing leak analysis on RISC-V processors at design level

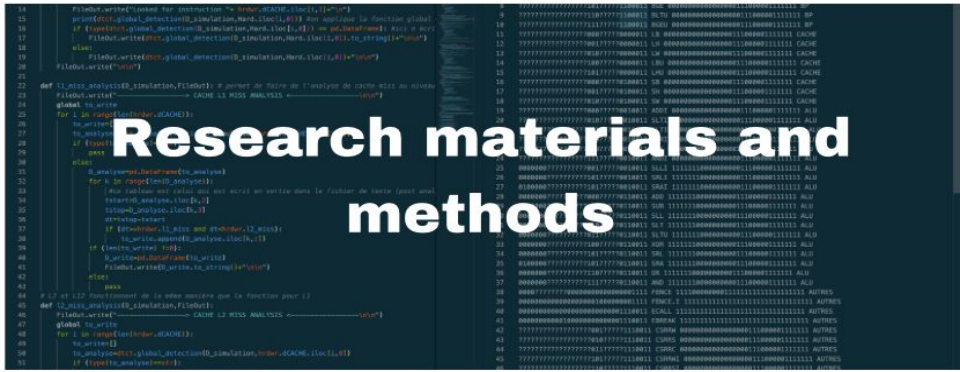
By Kenza Bouzergan, Moad Elhaddad, Bastian Krohg, Karine Marche, Claudia Menendez and Kilian Soual

4th year, Department of Electronics and Control Engineering, INSA - Toulouse, France
Tutor: **Mathieu Escouteloup**

LAAS CNRS

Outline


Background and context


Research materials and methods


Results

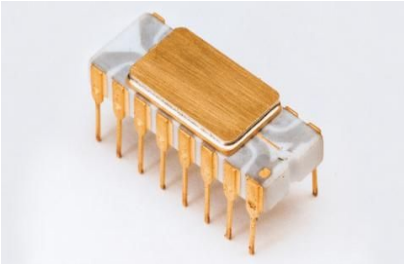

Discussion


Background and
context

1. Background and context

- Industry's **focus** on processor performance

- Moore's Law: number of transistors double almost every two years



1971:Intel 4004 2,300 transistors

2017 : I5-8600k Billions Of transistors

=> less secure systems!

1. Background and context

- Timing-based attacks
- Extract information by **measuring time differences** at runtime
- Operate on **hardware** - very problematic
- **Hardware**-targeting **timing leaks** in RISC-V processors during the design phase

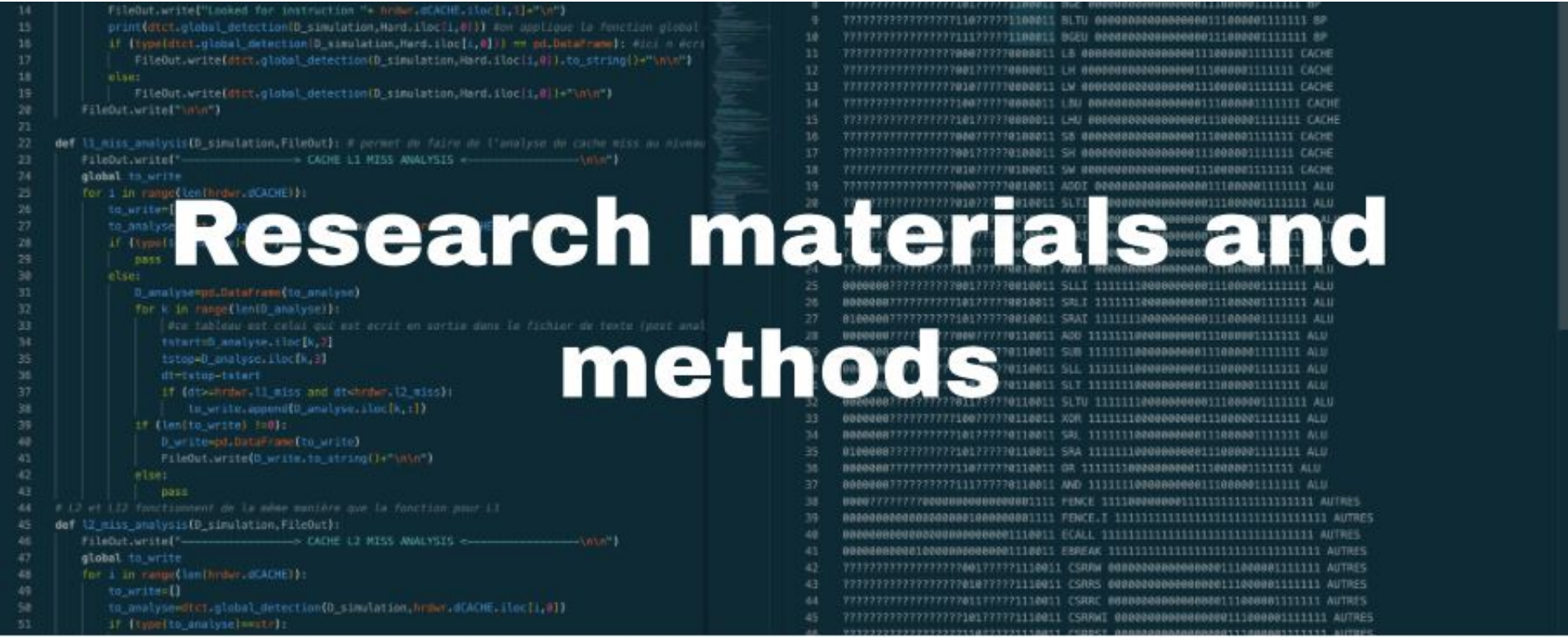
1. Background and context

RISC-V vs Modern processors

RISC-V **vs** intel

- Open source
- Simple Architecture

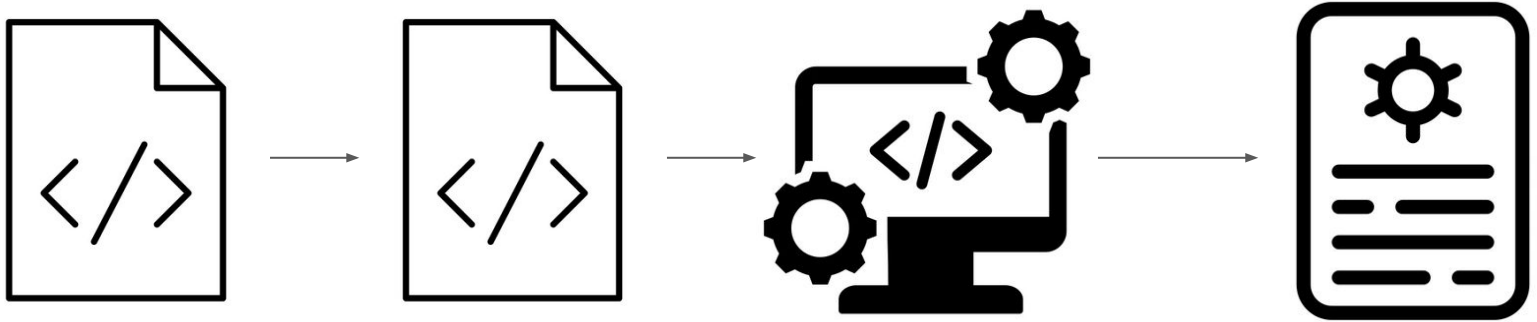
- Proprietary: Owned by / tied to a specific vendor



Research materials and methods

2. Research materials and methods

Exploitable trace generation



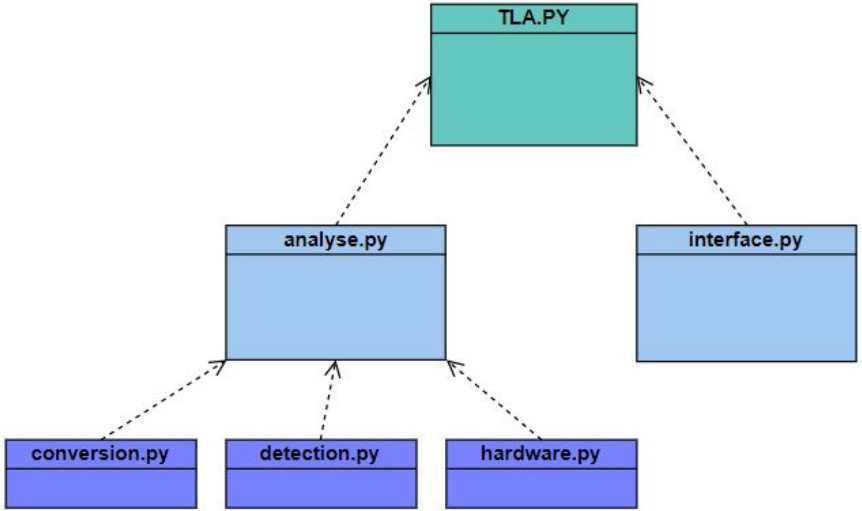
C- and RISC-V Assembly language programs

Hexadecimal format programs

Simulated RISC-V architecture processor

Instruction code trace file

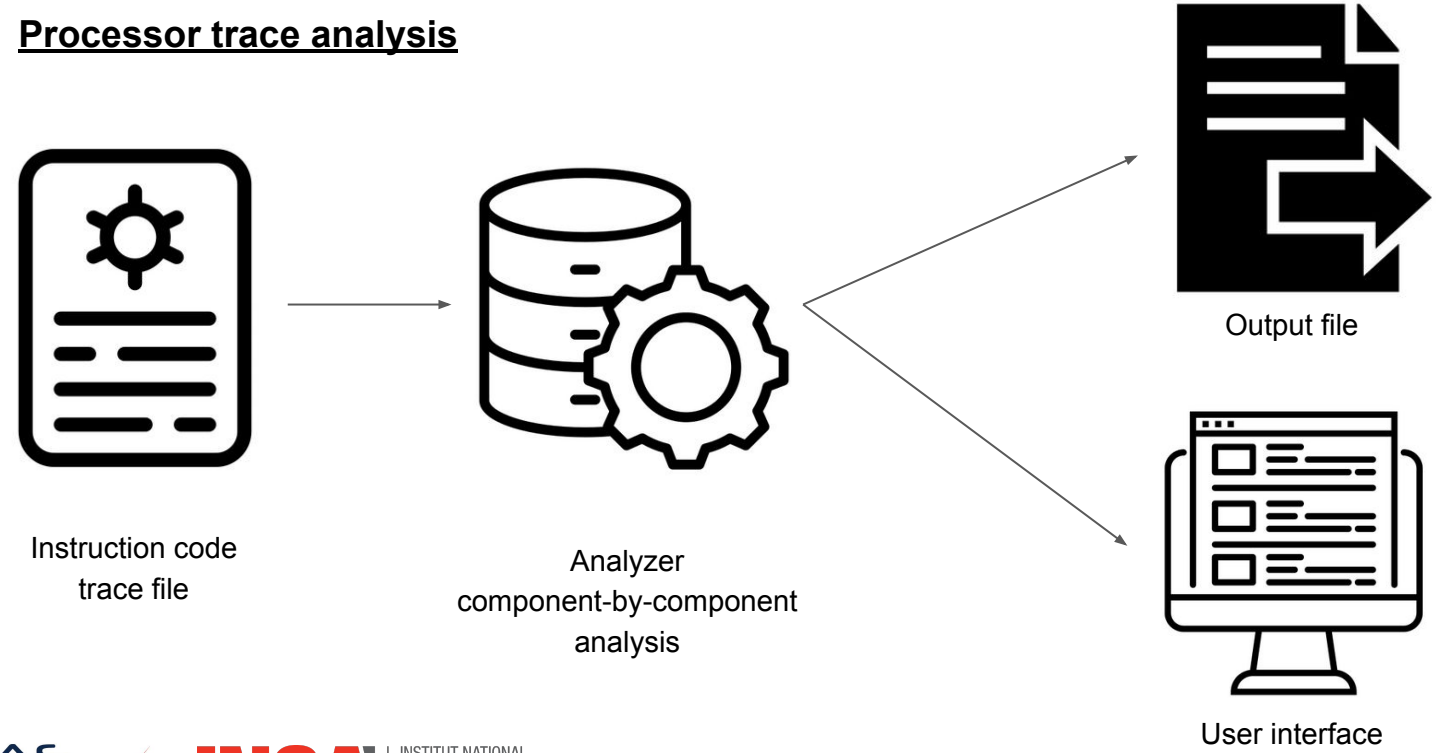
2. Research materials and methods



Overview of the analyzer's software architecture

2. Research materials and methods

Processor trace analysis



Results

3. Results

```
108 ------------------> CACHE L1 MISS ANALYSIS <------------------
109
110 ------------------> CACHE L2 MISS ANALYSIS <------------------
111
112             @ instruction_code  start_cycle  stop_cycle used_memory_@
113 10    00001020         1007a783           89         110      08000100
114 16    00001034         03c7a783          114         142      0800013c
115 23    0000104c         0001a203          149         170      08000000
116 28    0000104c         0001a203          173         194      08000020
117 33    0000104c         0001a203          197         218      08000040
118 38    0000104c         0001a203          221         242      08000060
119 43    0000104c         0001a203          245         266      08000080
120 48    0000104c         0001a203          269         290      080000a0
121 53    0000104c         0001a203          293         314      080000c0
```

Demonstration


Discussion
Sample text
Sample text
Sample text

4.Discussion

Some improvements are still needed…

- Developed a tool that classifies and organizes information for **manual** analysis to detect vulnerabilities.
- Future plans to **automate** the analysis process.
- Tool should be tested on physical RISC-V processors to validate its effectiveness in **real-world scenarios.**
- Need to consider "cache cold start" and "branch prediction cold start" phenomena when performing timing analysis.
- **Simplifications** were made in the initial version of the software, such as not accounting for dependencies between components during instruction execution.

4.Discussion

- Optimize our implementation
  - Reduce repetitive functions

- Implement matching pattern detection to detect attacks
  - Implement attack detection to ensure safety at design level

But it still is a good start for further projects!

5. Conclusion

- Tool developed for processor designers to **detect timing leaks** at the design level.
- 'Timing Leakage Analyzer' (TLA) in Python for analyzing trace files and generating reports.
- Current approach lacks cold start misses and dependency consideration, **requiring future improvements**.
- Reliable tool based on computed **True Positive Rates** and **timing criteria**.
- **Foundation for research** and enhancing hardware security against timing-based side channel attacks.

References

Outline images:

- *Literature Review*. literature-review.jpeg
- *Trust a study converted.* Trust-a-study-Converted.png
- *Analysis banner.* analysis-banner.jpg
- « What is Continuous Improvement (Kaizen) ? » , *The Lean Way*. https://theleanway.net/what-is-continuous-improvement
- « The Story of the Intel® 4004 » , *Intel*. https://www.intel.fr/content/www/fr/fr/history/museum-story-of-intel-4004.html
- ASCII Informatique - Magasin d'informatique à Nice, « INTEL Core i5 8600K - Socket 1151 - 6 Coeurs - 3.6/4.3Ghz - 9Mo » , *ASCII Informatique - Magasin d'informatique à Nice*. https://www.ascii-info.com/produit/pieces-detachees/processeurs/240/intel-core-i5-8600k-socket-1151-6-coeurs-3643ghz-9mo
- « La révolution RISC-V (1/4) : l'Agence tous RISC » , *MacGeneration*. https://www.macg.co/materiel/2023/03/la-revolution-risc-v-14-lagence-tous-risc-135597
- « Fichier : Intel logo (2006-2020).svg — Wikipédia » , 2 mars 2012. https://fr.m.wikipedia.org/wiki/Fichier:Intel_logo_%282006-2020%29.svg
- R. Malik, « ARM processors have a great decade ahead of them » . https://www.redsharknews.com/arm-processors-have-a-great-decade-ahead-of-them

Contacts

Kenza Bouzergan: bouzerga@insa-toulouse.fr

Moad Elhaddad: melhadda@insa-toulouse.fr

Bastian Krohg: krohg@insa-toulouse.fr

Karine Marche: marche@insa-toulouse.fr

Claudia Menendez: menendez@insa-toulouse.fr

Kilian Soual: soual@insa-toulouse.fr

Stay tuned to see our project on GitHub!

LAAS CNRS

INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES TOULOUSE