Secure Compilation of Counter-Measures against Spectre Attacks

City and country Nancy, France
Team and research laboratory Team Pesto, at Loria lab (Inria Nancy, CNRS, Université de Lorraine)
Name and address of the advisor Vincent Laporte, Vincent.Laporte@inria.fr

Context

The current practice for cryptographic implementations is to harden them against side-channel attacks and to this end ensure that they are *constant-time*. Unfortunately constant-time security does not protect against information leakage due to speculative execution, e.g., the Spectre vulnerability. Nonetheless, specific counter-measures can be efficiently deployed, such as speculative load hardening (A. Shivakumar et al. 2022).

Compilation and program optimization may interfere with counter-measures so that vulnerabilities might be unexpectedly introduced at compile-time in otherwise secure programs. Fortunately some optimizing compilers do preserve some security properties; for instance recent work (Barthe et al. 2020, 2021) has shown how to formally prove preservation of the constant-time property.

Objective of the internship

The aim of this internship is to understand how to formally justify that program transformations (such as the ones found in optimizing compilers) do preserve security against side-channel attacks, in spite of speculative execution.

The task of the intern would be to:

- study formal semantics allowing to describe speculative side-channel attacks;
- propose a proof technique to justify preservation of the corresponding security;
- apply said proof technique to realistic compilation passes.

Some or all of this work may be mechanized in the Coq proof assistant.

Bibliographic references

A. Shivakumar, Basavesh, Gilles Barthe, Benjamin Grégoire, Vincent Laporte, Tiago Oliveira, Swarn Priya, Peter Schwabe, and Lucas Tabary-Maujean. 2022. “Typing High-Speed Cryptography Against Spectre V1.” Cryptology ePrint Archive, Paper 2022/1270. https://eprint.iacr.org/2022/1270.

Barthe, Gilles, Sandrine Blazy, Benjamin Grégoire, Rémi Hutin, Vincent Laporte, David Pichardie, and Alix Trieu. 2020. “Formal Verification of a Constant-Time Preserving C Compiler.” *Proceedings of the ACM on Programming Languages* (*POPL*).

Barthe, Gilles, Benjamin Grégoire, Vincent Laporte, and Swarn Priya. 2021. “Structured Leakage and Applications to Cryptographic Constant-Time and Cost.” In *CCS 2021 - ACM SIGSAC Conference on Computer and Communications Security*, 462–76. CCS ’21: Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. Virtual Event, South Korea: ACM. https://doi.org/10.1145/3460120.3484761.

Expected ability of the student

The candidate should be familiar with formal semantics of programming languages. No knowledge in security and cryptography is required. Prior experience with the Coq proof assistant would be appreciated but is not mandatory.

Remarks

Possibility to continue as a PhD.