A Concolic Execution Engine for OTBN

Advisor: **Roderick Bloem**

Motivation

OpenTitan's ECC and RSA implementations run on OTBN, a RISC-V core with 256-bit integer instructions. Their secrets can leak through power or timing even when the algorithm is correct: a value that is unmasked mid-computation becomes visible in a power trace, and related secrets entering the ALU in successive cycles reveal something about their structure. Finding such points in several thousand instructions of assembly is done by hand today.

Symbolic execution can do it instead: lift the program to SMT formulas whose solutions witness a leak. Most of a cryptographic routine is not secret, though, so only a fraction needs symbolic treatment and the rest can stay concrete — concolic execution, which is what makes deep paths affordable.

Goals and Tasks

- Survey symbolic and concolic execution and pick an approach that suits OTBN
- Encode OTBN assembly in SMT so that satisfiability witnesses a leak
- Build the engine, standalone or on top of the existing OTBN simulator
- Run it on OpenTitan's crypto library and measure true and false positives

Literature

- R. Baldoni et al.
  A Survey of Symbolic Execution Techniques
  ACM Computing Surveys 2018
  doi:10.1145/3182657
- lowRISC
  OpenTitan Big Number Accelerator (OTBN)
  https://opentitan.org/book/hw/ip/otbn/

Courses & Deliverables

☑ **Master Project**
Project code
Report
Presentation

– OR –

☑ **Master's Thesis**
Initial presentation
Project code
Thesis (60+ pages)
Final presentation

Recommended if you're studying

☑ CS ☑ ICE ☑ SEM

Prerequisites

- Software security and static analysis
- SAT/SMT solvers (Z3); assembly

Advisor Contact

roderick.bloem@tugraz.at