Crypto Compiler

Advisor(s): David Derler and Daniel Slamanig

Institute for Applied Information Processing and Communications (IAIK)
Graz University of Technology, Austria

Motivation

When implementing cryptographic protocols, it is often a cumbersome and error-prone task to map concepts used in an abstract protocol specification to a concrete implementation.

Some supportive tools for this task have already been proposed. These tools can roughly be divided into two groups. Firstly, there are rapid prototyping tools that allow to specify protocols in an abstract high-level and easy to use language. The protocols can, then, be tested and run in the environment provided by the tool. Secondly, there are tools for generating code for a specific target language from abstract protocol definitions. Unfortunately, such tools typically define rather complicated specification languages.

We plan to launch a project to develop a modular and flexible tool, which combines these two approaches. That is, to implement a flexible and modular compiler that allows to translate some abstract high-level protocol specification language into pieces of source code of some target language (e.g., Java, C) while using cryptographic functionality from already existing libraries.

Goals and Tasks

Depending on your interest, you can choose to implement some module(s) for this project. Goals and tasks will then be defined in our first meetings.

Literature

- S. Meiklejohn et al.
  ZKPDL: A Language-Based System for Efficient Zero-Knowledge Proofs and Electronic Cash
  USENIX Security Symposium
  http://www.usenix.org/events/sec10/tech/full_papers/Meiklejohn.pdf
- J. A. Akinyele et al.
  Charm: a framework for rapidly prototyping cryptosystems
  Journal of Cryptographic Engineering 2013
- E. Bangerter et al.
  YAZKC: Yet another zero-knowledge compiler
  USENIX Security Symposium (Poster Session) 2010
- I. Damgård
  On Σ-protocols
  Tech. rep.
  http://www.cs.au.dk/~ivan/Sigma.pdf.
  2010
- M. Fredrikson and B. Livshits
  ZØ: An Optimizing Distributing Zero-knowledge Compiler
  USENIX Security Symposium

Studies

☒ INF ☒ TEL ☒ SW ☒ TM

Prerequisites

- Basic crypto knowledge
- Programming skills

Advisor / Contact

daniel.slamanig@iaik.tugraz.at