Empirical Evaluation of the Protocol Specification Language MSR 2

Rishav Bhowmick, Iliano Cervesato

Carnegie Mellon University – Qatar

iliano@cmu.edu

Background

- MSR 2
    - Protocol specification language
        - Strongly typed multiset rewriting with constraints
    - Designed in 2001
    - Used extensively in Kerberos project
            - F. Butler, A. Jaggard, A. Scedrov, J. Tsay, …
        - Experts
    - Implemented in 2004
            - M.-O. Stehr, S. Reich
        - Type-checking with type reconstruction
        - Execution (incl. limited search, tracking)
        - Constraints
- How usable is it by non-experts?

Project

- Student with no prior exposure
  - CMU-Q undergrad (sophomore)
    - No knowledge of security protocols
    - Programming experience in Java and C++
      - Otherwise rather sharp
  - Brief introduction on basic security
- Acquaintance to MSR 2
  - How hard is it to learn the paradigm?
- Encoding of the Clark-Jacob library
  - How hard is it to figure out the techniques?
  - How is the implementation performing?

Outcomes

- MSR language
  - Paradigm is easy to grasp (3 hours)
  - Techniques
    - Harder to figure out
    - Once figured out, sensible and easily replicable
- MSR implementation
  - Makes all the difference
  - Found several bugs
    - Type reconstruction is underpowered
    - Error messages are unhelpful
    - Lack of robustness
      - Inessential changes make insolvable constraints solvable

Future Work

- Fix implementation
- Low level protocol specification
- Explore linguistic feature to facilitate description of optional behaviors