Master Thesis Project

Uncovering State-Machine Bugs in Security Protocol Implementations

Contact: Ahmed Rezine[1] and Niklas Johansson[2]

[1] Linköping University
[2] Sectra

Implementations of security protocols, such as TLS or SSH, control and orchestrate message exchanges to account for the underlying protocol specification. As a consequence, such implementations need to carefully manage the type and the order of the messages. Deviations can result in serious vulnerabilities with failed connections or even in making possible security attacks. Testing implementations of security protocols is challenging because of their complexity, and because tests need to account for the history of the exchanges to capture meaningful scenarios. Recent works [1–3] propose to use automata learning approaches to build a finite state model of the protocol and to use it for black box testing of existing security protocols. This project will experiment with these approaches to assess their applicability in systematically testing security protocols. For this purpose, the project aims to:

1. Gain familiarity with testing of security protocols implementations.
2. Gain familiarity with the recent automata based testing of security protocols.
3. Experiment automata based testing on a security protocol.

Qualification

This 30 hp thesis will be carried by one or two master student(s).

- The student should have very good programming skills in C and Java
- The student should have taken a Networks Security course
- Having taken the software verification or the software testing course is a plus

References

1. Paul Fiterau-Brostean, Bengt Jonsson, Konstantinos Sagonas, and Fredrik Tåquist. Automata-based automated detection of state machine bugs in protocol implementations. In *NDSS*, 2023.
2. Paul Fiterău-Broştean, Bengt Jonsson, Konstantinos Sagonas, and Fredrik Tåquist. Smbugfinder: An automated framework for testing protocol implementations for state machine bugs. In *Proceedings of the 33rd ACM SIGSOFT International Symposium on Software Testing and Analysis*, ISSTA 2024, page 1866–1870, New York, NY, USA, 2024. Association for Computing Machinery.
3. Juraj Somorovsky. Systematic fuzzing and testing of tls libraries. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, page 1492–1504, New York, NY, USA, 2016. Association for Computing Machinery.