Software Engineering Seminar

Improving Fuzzing with Symbolic Execution

Description

Fuzzing is a great technique to, for example, discover and reproduce software system vulnerabilities. However, there exist problems with finding test inputs for complex checks (e.g., string equality checks). A recent approach proposes to combine fuzzing techniques with symbolic execution to effectively tackle this problem [1].

The student should examine and discuss the approach given in the paper and compare it with similar existing techniques.

References

[1] Nick Stephens, John Grosen, Christopher Salls, Andrew Dutcher, Ruoyu Wang, Jacopo Corbetta, Yan Shoshitaishvili, Christopher Krügel, and Giovanni Vigna. Driller: Augmenting fuzzing through selective symbolic execution. In *NDSS*, 2016.

Contacts

Simon Heiden (`heiden@informatik.hu-berlin.de`)
Software Engineering Group
Institut für Informatik
Humboldt-Universität zu Berlin