Software Engineering Seminar (WS 2016/17)

Automated Whitebox Fuzz Testing

Description

Fuzz testing is an effective technique for finding security vulnerabilities in software [1, 2, 3]. The idea is to combine symbolic execution and dynamic test generation to provide a suitable coverage of the code to detect problematic code fragments

The student is supposed to focus on automated whitebox fuzz testing and investigate the state of the art (approaches that also go beyond [1, 2, 3]).

Prerequisites

A basic knowledge of Software Engineering I/II .

References

[1] Patrice Godefroid, Michael Y. Levin, and David A. Molnar. Automated whitebox fuzz testing. In *Proceedings of the Network and Distributed System Security Symposium, NDSS 2008, San Diego, California, USA, 10th February - 13th February 2008*. The Internet Society, 2008.

[2] Patrice Godefroid, Michael Y. Levin, and David A. Molnar. SAGE: whitebox fuzzing for security testing. *Commun. ACM*, 55(3):40–44, 2012.

[3] Van-Thuan Pham, Marcel Böhme, and Abhik Roychoudhury. Model-based whitebox fuzzing for program binaries. In David Lo, Sven Apel, and Sarfraz Khurshid, editors, *Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering, ASE 2016, Singapore, September 3-7, 2016*, pages 543–553. ACM, 2016.

Contacts

Lars Grunske (`grunske@informatik.hu-berlin.de`)
Software Engineering Group
Institut für Informatik
Humboldt-Universität zu Berlin