STEVENS
INSTITUTE OF TECHNOLOGY
1870

Ph.D. DISSERTATION DEFENSE

Candidate:	Yifan Wang
Degree:	Doctor of Philosophy
School/Department:	Charles V. Schaefer, Jr. School of Engineering & Science / Computer Science
Date:	Tuesday, November 29th, 2022
Time/Location	3:00 p.m. (EST) / https://stevens.zoom.us/j/98840636085
Title:	Improving Efficiency, Effectiveness, and Evaluation of Fuzz Testing
Chairperson:	Dr. Georgios Portokalidis, Department of Computer Science, Stevens Institute of Technology
Committee Members:	Dr. Jun Xu, School of Computing, University of Utah Dr. Xueqing Liu, Department of Computer Science, Stevens Institute of Technology Dr. Hang Liu, Department of Electrical and Computer Engineering, Stevens Institute of Technology

ABSTRACT

Fuzzing, or fuzz testing, is a popular option for security testing. It works by continuously mutating existing test cases to produce new ones for exercising the target software. In recent years, fuzzing has gained tremendous development. However, it can still be limited when applied to real-world, complex software systems. This dissertation focuses on improving fuzzing in the security context from three critical aspects: efficiency, effectiveness, and evaluation.

High efficiency in covering different code regions --- and the vulnerabilities inside --- is a critically desired property of fuzzing. To augment the efficiency, a common strategy is parallel fuzzing, namely running many fuzzing instances in parallel to test the same target software. However, the parallel model of existing fuzzing tools runs all the instances naively without elaborate workload distribution. This can lead different instances to explore overlapped code regions, eventually offsetting the concurrency benefits. My dissertation research addresses this problem by developing a generic model to describe optimal parallel fuzzing. Following this model, my research further created a new solution, called AFL-EDGE, to realize optimized parallel fuzzing on top of mainstream fuzzing tools.

Low effectiveness in reaching and triggering vulnerabilities in software systems that process highly structured documents is a long-known barrier faced by today's fuzzing. To mitigate this barrier, my dissertation research introduces an intermediate document representation called DIR to describe a document file in an abstract way independent of the underlying format. My research further develops a series of multi-level, structure-ware mutations on the DIR to derive test cases to effectively reach and find vulnerabilities in document-processing software.

How to evaluate the capability of a fuzzing tool is also a major problem puzzling the security community. The conventional evaluation benchmarks focus on code coverage, failing to properly

unveil a fuzzing tool's expressiveness in finding vulnerabilities. To tackle this problem, my dissertation research proposes a new approach to produce vulnerability-driven benchmarks. The approach migrates heterogeneous, real-world vulnerabilities into the same software with fuzzing-related properties of those vulnerabilities maintained. Evaluating a fuzzing tool on such software, we can provide not only coarse-grained measurement results (e.g., the percentage of vulnerabilities the fuzzing tool can find) but also fine-grained understandings (e.g., the fuzzing tool is capable/incapable of finding vulnerabilities with certain properties).