Utvecklarstyrd fuzzing

Projektförslag till kandidatarbete, VT'23

Bakgrund

Fuzzing är en metod för att hitta buggar i mjukvara genom automatiskt indatagenerering som länge varit populärt för att hitta sårbarheter och på senare tid fått mycket uppmärksamhet inom akademin och industrin. Indatan kan genereras med flera olika tekniker, där några exempel är symbolisk exekvering och slumpmässig datagenerering. Målet med fuzzing är att hitta någon form av indata som leder till att programmet påvisar ett oönskat beteende.

Projektbeskrivning

Projektet har som mål att en utvecklare eller testare ska kunna använda en debugger för att sätta en brytpunkt någonstans i ett program. När villkoren för den brytpunkten nås används lokala tillstånd (variabler, funktionsargument, med flera) för att generera indata till en befintlig grammatikdriven fuzzer. Debuggern används sedan för att injicera och köra testfallen fuzzern genererar, samt för att inhämta data om systemeffekter såsom krascher.

Slutprodukten blir ett verktyg som kan användas för att generera indata baserat på en given grammatik.

Litteraturförslag

- *Fuzzing: Challenges and Reflections*, av Marcel Böhme; Cristian Cadar; och Abhik Roychoudhury.

  `https://mboehme.github.io/paper/IEEESoftware20.pdf`

  En lättläst introduktion till fuzzing som ger en snabb överblick över området.

- *The Fuzzing Book: Tools and Techniques for Generating Software Tests*, av Andreas Zeller, Rahul Gopinath, Marcel Böhme, Gordon Fraser, och Christian Holler.

  `https://fuzzingbook.org`

  En introduktionbok till fuzzing som finns fritt tillgänglig på bokens hemsida.

Förslagslämnare

Albin Otterhäll

Målgrupp

DV, D, IT

Speciella förkunskapskrav

Personerna som söker projektet bör ha ett intresse för IT-säkerhet. Det är en fördel om man har programmerat i C, men inget krav.

Handledare

Björn Ståhl, tekn.dr.