Brainstorm

June, 2020

XSym

- Loops
- Conditions and return values.

- Comparison with NAVEX
  - Target the same set of vuls and apps, find more exploitable paths and bugs
  - Evaluation: only exploit generation.

PHP type mismatch

- Same problem as TypeDevil (ICSE'15), but PHP instead. There was a poster paper about it.
- .htaccess

PHP built-in function type mismatch

- PHP built-in functions have excessive type inconsistency tolerance.
- Strcmp(array(), string) returns 0, denoting two arguments are equal.
- Using such kind of case to exploit a vulnerability is rare.

State-aware fuzzing

- Complicated applications have many states
  - E.g., in web applications, *index.php->login.php->view.php*
  - USENIX'12 *"Enemy of the State: A State-Aware Black-Box Web Vulnerability"*
  - Navigation graph (dynamic part) in NAVEX is also a state-aware fuzzing
- Whether state-aware helps improve kernel fuzzing?
  - E.g., in file system fuzzing, to generate a series of context-aware system calls, it also maintains such "state" of contexts?

Concurrency bugs

- Concurrency bugs have been investigated in kernels, file systems, and web applications.
- ICSE'12: Web applications do not have explicit primitives (e.g., lock, mutex, etc.) like in kernels and file systems. They have resource related concurrency bugs, e.g., database access, file system access, etc.
- *Solution: Transaction database*
- Types of concurrency bugs
  - Deadlocks
  - Non-deadlocks: atomicity violation, order violation, data race
- An empirical study of web concurrency bugs?
- How about run-time defense for web concurrency bugs?
  - *FSE'12 "AI: A Lightweight System for Tolerating Concurrency Bugs"*
  - *What is a successful defense/mitigation?*

Program analysis in program execution engines

- There are fuzzing tools for JavaScript engine recently.
  - USENIX'20, S&P'20
- How about PHP engine, Python interpreter?
- How about logic bugs that do not crash the engines? How to define "correctness" of execution results in the execution engine?

Program analysis for Go/Rust

- Go and Rust are quite new programming languages.
- There are not many tools.
  - Detecting concurrency bugs in Go/Rust?