Your Browser is my Fuzzer: Fuzzing Native Applications in Web Browsers

Jonathan Metzman
metzman@chromium.org
@metzmanj

whoami

- Chrome Security Fuzzing Team
  - Representing myself, not Google.
- ClusterFuzz and OSS-Fuzz

In-Browser Coverage Guided Fuzzing

- Build ordinary native code for use in the browser (**WebAssembly/WASM**)
- Fuzz it with libFuzzer

Fuzzing in the Browser!

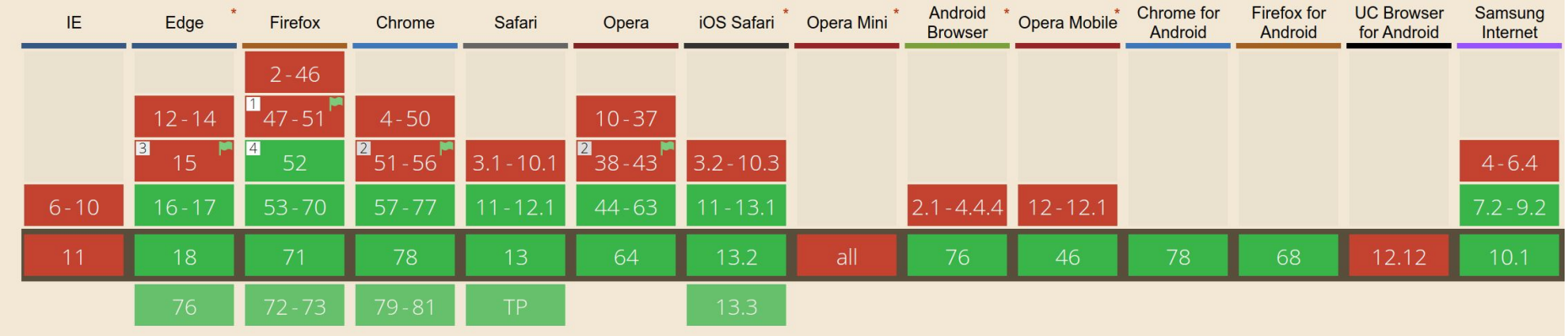
SQLite Demo

Crash Demo


```
int LLVMFuzzerTestOneInput(const uint8_t *data, size_t size) {
  if (size > 0 && data[0] == 'H')
    if (size > 1 && data[1] == 'I')
      if (size > 2 && data[2] == '!') {
        uint8_t* p = (uint8_t*) malloc(10);
        free(p);
        return p[0];
      }
  return 0;
}
```

tinyurl.com/libfuzzer-wasm

Browser Support



IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Opera Mobile *	Chrome for Android	Firefox for Android	UC Browser for Android	Samsung Internet
		2-46											
	12-14	1 47-51	4-50		10-37								
	3 15	4 52	2 51-56	3.1-10.1	2 38-43	3.2-10.3							4-6.4
6-10	16-17	53-70	57-77	11-12.1	44-63	11-13.1		2.1-4.4.4	12-12.1				7.2-9.2
11	18	71	78	13	64	13.2	all	76	46	78	68	12.12	10.1
	76	72-73	79-81	TP		13.3							

Why?

OSS-Fuzz@Home

A Whole New World of Vulnerabilities

- Important WebAssembly apps are coming

Virtual Machine

How?

WebAssembly (WASM)

- Instruction format/virtual machine
- Supported by all important browsers
- Near-native speed

emscripten

Emscripten is a toolchain for compiling to asm.js and WebAssembly, built using LLVM, that lets you run C and C++ on the web at near-native speed without plugins.

Porting

Compile your existing projects written in C or C++ and run them on all modern browsers.

APIs

Emscripten converts OpenGL into WebGL, and lets you use familiar APIs like SDL, or HTML5 directly.

Fast

Thanks to LLVM, Emscripten, asm.js and WebAssembly, code runs at near-native speed.

LLVM Based Fuzzing Tools

- LibFuzzer
- And Friends!
  - ASAN
  - UBSAN

Can I haz in-browser fuzzing?

- Guide
- Steps
  - Compile project with emscripten and -fsanitize-coverage=inline-8bit-counters
  - Link against libFuzzer.

Near Native Speed

- Modified libFuzzer to be a cooler demo
- Real Speed (sqlite-fast)

```
REDUCE cov: 799 ft: 1314 corp: 277/6455b lim: 1480 exec/s: 40361
72 MS: 3 EraseBytes-CopyPart-CopyPart-
REDUCE cov: 799 ft: 1314 corp: 277/6454b lim: 1480 exec/s: 40428
172 MS: 1 EraseBytes-
REDUCE cov: 801 ft: 1317 corp: 278/6459b lim: 1490 exec/s: 40701
```

Future Plans

- First class support
  - -fsanitize=fuzzer
- Binary only fuzzing
- Performance

Summary

- Native code can run in web browsers using emscripten + WASM
- Fuzz it using libFuzzer/ASAN/UBSAN
- Figure out something awesome to do with this!

Links

- Demo
  - tinyurl.com/libfuzzer-wasm
- Guide
  - github.com/jonathanmetzman/wasm-fuzzing-demo
- Emscripten (emscripten.org)

Questions?

Backup Slides

Missing Features

- RSS Limit
  - Relies on threads
    - Only Chrome supports
  - Spectre/Meltdown ruined everything!
  - Less important in-browser
- Timeouts
  - setitimer not supported