

Yuyang(Peter) RONG

(+1)530 · 601 · 3646 ◊ PeterRong96@gmail.com ◊ [Webpage](#) ◊ [Github](#)

EDUCATION

UC Davis

Sep 2019 - Dec 2022

Ph.D. candidate in Computer Science

Davis, CA

- Research interests: **fuzzing, software security, compiler testing, program analysis**
- Languages: **C/C++ (9/10), Rust (8/10), Python, Java**
- Tools: **LLVM (LLVM contributor), CMake, Docker, Linux, AFL++**

ShanghaiTech University

Sep 2015 - Jun 2019

B.E. Computer Science and Technology

Shanghai, China

- GPA 3.79/4 (Ranking: 5/124)
- Excellent Undergraduate of Shanghai (Jun 2019) Scholarship of Academic Excellence (Nov 2017)

EXPERIENCE

Advanced Micro Devices (AMD)

Jun 2022 - Dec 2022

Research Intern

San Jose, CA

- Focused on testing compiler backend of AI Engine.
- Implemented a state-of-the-art fuzzer to accommodate for the compiler infrastructure, [open-sourced](#).
- Found over 40 missing features in AI Engine. Found 54 confirmed bugs in LLVM upstream, 26 fixed, [bug tracker](#).
- Lightning talk accepted to [2022 LLVM Developer's Meeting](#), [recording](#).

Bytedance

Jun 2020 - Sep 2020

Research Intern

Mountain View, CA

- Focused on optimizing fuzzer Angora's gradient solver and alleviate branch collision problem.
- Implemented an LLVM pass in 2000 lines of C++ and a new gradient solver in 3000 lines of Rust, [open-sourced](#).
- Improved branch coverage by 41% compared by Angora, 94% compare to AFL++.

Bytedance

Sep 2018 - Aug 2019

Research Intern

Beijing, China

- Assigned to find integer errors using Angora in Bytedance's codebase.
- Designed a sanitizer and implemented it as an LLVM pass with runtime library using around 1500 lines C++ and 2000 lines of Rust, maintainer of Angora ever since.
- Identified 8 crashing errors that could cause denial of service attack, [CVE-2020-18869](#) and [CVE-2020-18871](#) assigned; found 166 non-crashing errors that could cause program misbehave, reported to developers.

SELECTED PUBLICATIONS

- Understanding Programs by Exploiting Fuzzing Test Cases (ACL 2023, submitted)
- IrFuzzer: Specialized Fuzzing for LLVM Backend Code Generation (S&P 2023, submitted)
- Valkyrie: Improving Fuzzing Performance Through Principled Techniques (QRS 2022, best paper award)
- IntEgrity: finding integer errors by targeted fuzzing (SecureComm 2020)
- An inexact first-order method for constrained nonlinear optimization (Optimization Methods and Software 2022)