Hao Sun

(+86) 13220190100
Haidian, Beijing, China
sunhao.th@gmail.com

M.Sc., Tsinghua, System Security

GitHub: SunHao-0
Page: haosun.info

- System security researcher, focus on operating system kernel, **Linux** kernel mostly.
- Interested in developing practical vulnerability discovery tools, author of kernel fuzzer **Healer**.
- Passionate about designing complex, distributed systems, familiar with source code of multiple huge systems.
- Beginner fan of programming language theory, preferred language is **rust** currently.

Education

- **M.Sc.**, *School of Software*, *Tsinghua University (THU)* 2020 — Current
  Software System Security Assurance Group, supervised by Prof.Yu Jiang.
  System security, Operating system Kernel

- **B.Sc**, *School of Software*, *Beijing University of Posts and Telecommunications (BUPT)* 2016 — 2020

Publications

- **HEALER: Relation Learning Guided Kernel Fuzzing**
  Hao Sun, Yuheng Shen, Cong Wang, Jianzhong Liu, Yu Jiang, Ting Chen, and Aiguo Cui
  *2021, ACM SIGOPS 28th Symposium on Operating Systems Principles (**SOSP '21**)*

- **Rtkaller: State-aware Task Generation for RTOS Fuzzing**
  Yuheng Shen , **Hao Sun**, Yu Jiang, Heyuan Shi, Yixiao Yang, and Wanli Chang
  *2021, ACM Transactions on Embedded Computing Systems (**EMSOFT '21**)*

- **Go-Sanitizer: Bug-Oriented Assertion Generation for Golang**
  Cong Wang , **Hao Sun**, Yiwen Xu, Yu Jiang, Huafeng Zhang, Ming Gu
  *2019 IEEE International Symposium on Software Reliability Engineering Workshops (**ISSREW**)*

Software

- **Healer, kernel fuzzer inspired by Syzkaller**
  Healer is an automated kernel vulnerability discovery tool, written in 17,000+ lines of pure rust. It utilizes syscall specifications, which encode structural and partial semantics of input, to generate syscall sequences. Assisted with various sanitizers, Healer detects kernel bugs via triggering kernel crashes with the generated sequences. The idea of Healer is pretty straightforward, but it incorporates many hacking techniques for efficient implementation, and eventually simulates a running environment with high system load, which explains its effectiveness.

  PARTIAL RESULTS: 100+ reported and fixed Linux bugs, **10+** CVEs assigned, **187** stars on github.

- **KSG, kernel syscall specification generator**
  Writing syscall specifications in domain language requires huge efforts while being laborious, KSG is designed to generate them automatically for *Healer* and Google *Syzkaller*. It incorporates probe-based tracing, symbolic execution-based analysis to extract syscall information from source code. KSG is implemented with 7000+ lines of C++ code based Clang Static Analyzer (CSA). It has been submitted to **ATC' 22** but has not been published yet.

- **UFUZZ, fuzzer for OSEK/VDX RTOS kernel.**
  UFUZZ is an automated bug discovery tool, designed for embedded RTOS kernel that conforms to OSEK/VDX specification and implemented with pure rust. It generates test cases with the awareness of the application model, e.g., prioritized tasks and ISRs, efficiently transfers inputs via directly accessing the memory of guest VM, evolves corpus with execution feedback collected from t32 simulator. UFUZZ has been adopted and deployed in a private organization.

Experience

- **Intern**, System Developing, *Architecture Group*, *ByteDance Ltd*. 01/2019 — 06/2019
  Designed and implemented a core data hub system with 10k+ QPS in Golang. Based on a Graphql-liked framework, the hub separated data fetching and extracting into different components as well as cached hot data into LRU buffer thus increasing system throughout rate and stability.

- **Intern**, System Developing, *Cainiao Network*, *Alibaba Group*. 5/2018 — 9/2018
  Designed and implemented a template-based database interacting code generator, which automatically generates code that interacts with the database in a uniform style and reduces manual efforts.