

Hongxu Chen | Curriculum Vitae

☎ +6585476746 • ✉ hongxu.chen@ntu.edu.sg
🌐 hongxuchen.github.io • 🌐 HongxuChen • 🐦 hongxuchen
in hongxu-chen-09a97640

Education

Nanyang Technological University

Singapore

Ph.D., Major in Cybersecurity, Supervisor: Prof. Yang Liu

2015.08 ~ 2019.07

Thesis: Securing Software Systems via Fuzz Testing and Verification

o **Fuzz Testing:** Grey-box fuzzing on C/C++ programs with the help of program analysis.

o **Formal Verification:** Permission-dependent type system for secure information flow analysis.

Shanghai Jiaotong University

China

Master, Major in Program Analysis, Supervisor: Prof. Jianjun Zhao

2011.09 ~ 2014.03

Thesis: Program Slicing Enhanced Symbolic Execution

Nanjing University of Science and Technology

China

Bachelor, Major in Computing Mathematics

2007.09 ~ 2011.07

Working Experience

Research Fellow

Singapore

Nanyang Technological University

2020.01 ~ Present

I lead the architecture design of a fuzzing service which provides a general-purpose interface for different testing scenarios, such as embedded systems, the automotive-relevant microcontrollers, stateful protocols, etc. I am also involved in a data-based reverse engineering research to help decompose the functional components of a given project.

Research Associate

Singapore

Nanyang Technological University

2019.08 ~ 2019.12

I maintain the fuzzing framework FOT. Meanwhile, I am also involved in a high-performance cross-CPU binary fuzzing framework BiFF, as well as a fuzzing technique called MUZZ that aims to boost fuzz testing on multithreaded programs.

Research Intern

Singapore

Scantist

2018.09 ~ 2019.05

I co-designed and built the prototype of a cross-CPU grey-box fuzzer for binaries.

Research Associate

Singapore

Nanyang Technological University

2014.05 ~ 2015.08

I focused on LLVM based data flow analysis which aims to improve the dynamic fuzzing effectiveness with the aid of static analysis.

Research Intern

China

Microsoft Research Asia

2013.02 ~ 2014.11

I focused on improving the white-box fuzzing technique on patching programs with the help of static analysis; I implemented a static analysis tool that can slice the underlying program for subsequent white-box testing.

Research Projects

FOT: 2017.07 ~ present I develop and maintain the grey-box fuzzing framework FOT (Fuzzing Orchestration Toolkit). This framework facilitates static analysis to improve overall fuzzing effectiveness. FOT has integrated several existing and we have proposed new fuzzing techniques based on it.

o Project site: <https://sites.google.com/view/fot-the-fuzzer>.

o FOT has been successfully detecting 300+ vulnerabilities in 120+ open source projects. Among the detected zero-day vulnerabilities, 61 have been assigned with CVE IDs, including 10 with critical or high severity according to CVSS3.0. The vulnerability details are available at <https://github.com/ntu-sec/pocs>.

o FOT received 1st award in NASAC 2017 prototype competition (fixed topic) and was accepted by ESEC/FSE 2018; Another two fuzzing techniques based on FOT, Hawkeye and Cerebro, were accepted by CCS 2018 and ESEC/FSE 2019 respectively.

BiFF: 2018.11 ~ present I am involved in the development of high-performance cross-CPU binary fuzzing framework BiFF, which aims to improve the existing binary-only fuzzing techniques. BiFF facilitates our self-designed hooking technique and optimizes the fuzzing flow for service-like applications, and boosts overall performance of fuzzing against IoT devices with different CPU architectures. BiFF received 1st award in NASAC 2019 prototype competition (freestyle).

Hawkeye: 2017.12 ~ 2018.05 I designed the directed grey-box fuzzing technique Hawkeye, proposed the four properties a directed fuzzer is supposed to possess and provided our solutions; we experimentally demonstrated the effectiveness of Hawkeye. This work was accepted by CCS 2018.

STAndroid: 2015.08 ~ 2017.06 This project was inspired by the Android permission mechanism and we apply a secure type system to stress a category of information security problems. I designed the type system and proved the soundness. I implemented a checking tool to detect information leakage based on this type system. This work was accepted by CSF 2018.

RBScope: 2013.02 ~ 2013.11 This project aimed to apply static analysis to improve the effectiveness of the white-box fuzzing. The idea is to prune irrelevant program segments to reduce the search space of the symbolic execution testing technique. I implemented the program slicing based on LLVM framework.

Awards

1st Award in Prototype Competition (freestyle)	China
<i>The 18th National Software Application Conference (NASAC 2019)</i>	<i>2019.11</i>

1st Award in Prototype Competition (fixed topic)	China
<i>The 16th National Software Application Conference (NASAC 2017)</i>	<i>2017.11</i>

NTU Research Scholarship	Singapore
<i>Nanyang Technological University</i>	<i>2015.08 ~ 2019.07</i>

Publications

[#] indicates the top tier A+ conference/journal. [#] indicates the top tier A conference/journal.

1. [#] **Hongxu Chen**, Shengjian Guo, Yinxing Xue, Yulei Sui, Cen Zhang, Yuekang Li, Haijun Wang, and Yang Liu. MUZZ : Thread-aware grey-box fuzzing for effective bug hunting in multithreaded programs, The 29th Usenix Security Symposium (Usenix Security 2020), Boston, MA, USA,

August 2020. (Acceptance rate: $157/977 = 16.1\%$).

2. [#] **Hongxu Chen**, Yinxing Xue, Yuekang Li, Bihuan Chen, Xiaofei Xie, Xiuheng Wu, and Yang Liu. *Hawkeye: Towards a Desired Directed Grey-box Fuzzer*, the 25th ACM Conference on Computer and Communications Security (CCS 2018), pp. 2095–2108, Toronto, Canada, Oct. 2018. (Acceptance rate: $134/809 = 16.6\%$).
3. [#] **Hongxu Chen**, Yuekang Li, Bihuan Chen, Yinxing Xue and Yang Liu, *FOT: A Versatile, Configurable, Extensible Fuzzing Framework*, The 26th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2018), pp. 867–870, Lake Buena Vista, Florida, USA, Nov. 2018.
4. [#] **Hongxu Chen**, Alwen Tiu, Zhiwu Xu and Yang Liu, *A Permission-Dependent Type System for Secure Information Flow Analysis*, The 31st IEEE Computer Security Foundations Symposium (CSF 2018), pp. 218–232, Oxford, UK, Jul. 2018. (Acceptance rate: $34\% = 25/72$)
5. [#] Cheng Wen, Haijun Wang, Yuekang Li, Shengchao Qin, Yang Liu, Zhiwu Xu, **Hongxu Chen**, Xiaofei Xie, Geguang Pu, and Ting Liu. *Memlock: Memory usage guided fuzzing*. The 42nd International Conference on Software Engineering (ICSE 2020) (*accepted*), Seoul, South Korea, May 2020.
6. [#] Haijun Wang, Xiaofei Xie, Yi Li, Cheng Wen, Yang Liu, Shengchao Qin, **Hongxu Chen**, and Yulei Sui. *Typestate-guided fuzzer for discovering use-after-free vulnerabilities*. The 42nd International Conference on Software Engineering (ICSE 2020) (*accepted*), Seoul, South Korea, May 2020.
7. [#] Xiaofei Xie, **Hongxu Chen**, Yi Li, Ma Lei, Yang Liu, and Jianjun Zhao. *Deephunter: A coverage-guided fuzzer for deep neural networks*. The 34th IEEE/ACM International Conference on Automated Software Engineering (ASE 2019), San Diego, California, USA, Nov. 2019.
8. [#] Yuekang Li, Yinxing Xue, **Hongxu Chen**, Xiuheng Wu, Cen Zhang, Xiaofei Xie, Haijun Wang and Yang Liu. *Cerebro: Context-aware Adaptive Fuzzing for Effective Vulnerability Detection*. 27th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2019), Tallinn, Estonia, Aug. 2019. (Acceptance rate: $74/303=24\%$)
9. [#] Xiaofei Xie, Lei Ma, Felix Juefei-Xu, Minhui Xue, **Hongxu Chen**, Yang Liu, Jianjun Zhao, Bo Li, Jianxiong Yin and Simon See. *DeepHunter: A Coverage-Guided Fuzz Testing Framework for Deep Neural Networks*. the 28th International Symposium on Software Testing and Analysis (ISSTA 2019), Beijing, China, Jul. 2019. (Acceptance rate: $32/134=23.8\%$)
10. [#] Yinxing Xue, Guozhu Meng, Yang Liu, Tian Huat Tan, **Hongxu Chen**, Jun Sun, and Jie Zhang. *Auditing Anti-Malware Tools by Evolving Android Malware and Dynamic Loading Technique*. IEEE Transactions on Information Forensics & Security (TIFS), 12(7): 1529–1544, Jul. 2017. (IF 6.211).
11. [#] Xiaofei Xie, Yang Liu, Wei Le, Xiaohong Li, and **Hongxu Chen**. *S-Looper: Automatic Summarization for Multipath String Loops*. International Symposium on Software Testing and Analysis (ISSTA 2015), pp. 188–198, Baltimore, MD, USA, Jul. 2015. (Acceptance rate $33/117=27.7\%$)

Professional Skills

Proficient: Static Program Analysis, Grey-box Fuzzing, Symbolic Execution, Binary Analysis, Java, Python, Rust, Scala

Familiar: Program Language Theory, Compiler Techniques, Linux System Programming, Formal

Verification, LLVM/GCC, Bash, JVM, C/C++
Knowledgeable: OCaml, Haskell, Coq, Isabelle

Teaching Experience

Object-Oriented Programming: Autumn Semester 2018, NTU Lab supervision for course “CE/CZ2002 Object Oriented Design and Programming”.

Software Engineering: Spring Semester 2018, NTU Lab supervision for course “CE/CZ2006 Software Engineering”.

System Design and Programming: Autumn Semester 2017, NTU Lab supervision for course “CE/CZ3003 Software Systems Analysis and Design”.

Computer Security: Autumn Semester 2017, NTU Course design for “CE/CZ4062 Computer Security”.

Computer Security: Spring Semester 2017, NTU Lab supervision for course “CE/CZ4024 Cryptography and Network Security”.

Algorithms: Autumn Semester 2016, NTU Lab supervision for course “CE/CZ2001 Algorithms”.

Compiler Techniques: Spring Semester 2016, NTU Lab supervision for course “CE/CZ3007 Compiler Techniques”.