Qiushi Wu

Email : wu000273@umn.edu

Address: 1024 27th Ave SE Minneapolis, MN 55414 Mobile: +1-651-208-2318

Education

- **University of Minnesota, Twin Cities** Minneapolis, MN
  Doctor of Science in Computer Science & Engineering (Degree expected in 2023) *Sep. 2018 - Present*
- **University of Science and Technology of China** Hefei, China
  Bachelor of Engineering in Information Security *Sep. 2013 - July. 2018*

Research Interests

- **Systems Security:** Protect the security of widely used systems and software from semantic bugs, vulnerabilities, and insecure designs
- **Program Analysis:** Develop program-analysis techniques and tools to detect security-related issues in programs
- **Natural Language Processing (NLP):** Leverage NLP techniques to mine the text information in programs and patches to facilitate the program analysis

Publications & On Submission Works & Their contributions to the field and society

* Top-tier conferences in security: NDSS, USENIX Security, IEEE S&P

- **Qiushi Wu**, and Kangjie Lu "On the Feasibility of Stealthily Introducing Vulnerabilities in Open-Source Software via Hypocrite Commits" 42nd IEEE Symposium on Security and Privacy (**Oakland'21**), 2021.
  - A study of malicious-committer capabilities for open-source software (OSS).
  - A new vulnerability-introducing approach with increased stealthiness.
  - Discussion and suggestions for mitigating the risks for OSS communities.
- **Qiushi Wu**, Aditya Pakki, Navid Emamdoost, Stephen McCamant, and Kangjie Lu. "Detecting Disordered Error Handling with Precise Function Pairing." *30th USENIX Security Symposium (**USENIX Security'21**),* 2021.
  - A new class of error-handling bugs: Disordered Error Handling.
  - Precise function pairing technique.
  - An effective detection system.
  - Found hundreds of new security bugs in the Linux kernel, the FreeBSD kernel, and OpenSSL.
- Navid Emamdoost, **Qiushi Wu**, Kangjie Lu, and Stephen McCamant, "Practically Detecting Kernel Memory Leaks in Specialized Modules and Beyond." Conditionally accepted by *The Network and Distributed System Security Symposium (**NDSS'21**),* 2021.
  - An approach for identifying specialized memory allocation/deallocation functions.
  - An ownership reasoning mechanism for kernel objects.
  - A scalable implementation and numerous new bugs with 41 CVE assigned.
- **Qiushi Wu**, Yang He, Stephen McCamant, and Kangjie Lu. "Precisely Characterizing Security Impact in a Flood of Patches via Symbolic Rule Comparison." *The Network and Distributed System Security Symposium (**NDSS'20**),* 2020.
  - Symbolic rule comparison for automatically determining security impacts of bugs.
  - Finding of security bugs and unpatched vulnerabilities in Android OS.
- Kangjie Lu, Aditya Pakki, and **Qiushi Wu**. "Detecting missing-check bugs via semantic-and context-aware criticalness and constraints inferences." *28th USENIX Security Symposium (**USENIX Security'19**),* 2019.
  - A new system for missing-check bug detection.
  - Multiple new general techniques including automated critical-variable inference, two-layer type analysis, and cross-checking.
  - Finding numerous new bugs in the Linux kernel.
- Kangjie Lu, Aditya Pakki, and **Qiushi Wu**. "Automatically Identifying Security Checks for Detecting Kernel Semantic Bugs." *European Symposium on Research in Computer Security (**ESORICS'19**)*, 2019.
  - Automatic identification of security checks.

- Detection of three classes of semantic bugs.

- Bowen Wang, **Qiushi Wu**, Aditya Pakki, and Kangjie Lu, "Unleashing Fuzzing Through Comprehensive, Efficient, and Faithful Exploitable-Bug Exposing." Sumitted to **IEEE TDSC'20**.
  - A co-design of dual-execution and fuzzers that ensures fuzzing efficiency.
  - Multiple techniques, including practical and deterministic dual-execution engine, bug-sensitive diversification, comprehensive (both control-flow and data-flow) and efficient divergence detection.

Ongoing Projects

- **OS Kernel Bug Detection:** Detecting security-related bugs introduced by API design problems in the Linux kernel through code analysis and machine learning.
- **Open-Source Security:** Studying how vulnerabilities can be introduced in open source programs by seemingly valid patches.
- **Vulnerability Impact Analysis:** Analyzing the known CVE vulnerabilities through programming analysis and NLP to identify drawbacks of the current Common Vulnerability Scoring System (CVSS)

Experience & Professional Activities

- **Teaching Experience** University of Minnesota, MN
  Teaching assistant *Sep. 2018 - Jan 2019*
  - **CSCI 2021**: Machine Architecture and Organization
- **Research Experience** University of Minnesota, MN
  Research assistant in system-security lab, advised by professor Kangjie Lu *Jan 2019 - Present*
  - **Paper reviewer**: Helped to review papers in CCS 2019, ICICS 19, CCS 2020, and NDSS 2021
  - **Open-source contributor**: Reported and patched hundred of bugs in opensource projects
  - **Vulnerability identifier**: Identified dozens of CVE vulnerabilities in the Linux kernel

Technical Skills

- **Languages:** C, C++, Python, Shell, Matlab, SQL, Java, HTML
- **Working knowledge:** Linux, Windows, Vim, Git, Microsoft Office