Navid Emamdoost **Last updated on February 25, 2021**

Contact and Education Information

Navid Emamdoost
Cell: +1-612-666-7976
emamd001@umn.edu

PhD Candidate in Computer Science
University of Minnesota
Expected: May 2021

Interests

Applications of source-level and binary-level Program Analyses to software security. Including: **Binary Hardening and Isolation**, **Symbolic Execution**, **Information Flow and Taint Analysis**.

Professional Experience

Mozilla, Portland, OR, USA

Research Intern **May 2018 - Aug 2018**

- **Bringing Dynamic Loading into WebAssembly**
  Design and implementation of a dynamic loading library for **WebAssembly**. In this project, we showed how a **Rust** project which targets WebAssembly may have lazy-loaded modules that can be downloaded and instantiated at runtime. Functions in lazy modules are accessed via wasm table entries and indirect call instructions.

Squigl, Minneapolis, MN, USA

Software Engineering Intern **Jun 2016 - Sep 2016**

- **Enhancing the Animation Generation Software**
  Added new features like inline video, background, overlay and text insertion to the animation generation software using FFMPEG multimedia library in **.NET** framework.

Academic Research Experience

University of Minnesota, MN, USA

Research Assistant **Sep 2013 - present**

- **Automatic Semantic Error Detection in Kernel**
  Developed an **LLVM**-based static analysis tool to detect multiple classes of security bugs in the Linux kernel code. Found over **130** confirmed security bugs and received over **40 CVEs** for the detected vulnerabilities including **Use-After-Free**, **Null-Pointer-Dereference**, and **Memory-Leak**. The provided patches to the Linux maintainers are integrated into the upstream kernel code.
- **Software-based Fault Isolation**
  Improved runtime performance of Google Native Client (NaCl), specifically via reducing instruction padding overhead. Changed the NaCl instruction padding in GNU **Assembler** (GAS), updated NaCl validator to enforce security policies, and proved the validator correctness in **Coq**.
  Used OProfile and **Valgrind** to analyze the compiled binary runtime performance.
- **Quantitative Information Flow Analysis**
  Quantitatively measuring the information leaked by a program's output from the secret input. Used a combined approach of bit-level **taint analysis** and **symbolic execution**. Critical information flow regions are identified via taint analysis, and then these regions of code are symbolically executed to measure information influences.
- **Binary Mutation for Test Analysis**
  Evaluating the adequacy of a test suite, via **static binary rewriting**. Used binary mutation to generate different mutants via instruction rewriting. Such

mutants were used to measure the quality of the test suite. The project demonstrated how binary mutation is effective in test quality measurement when no source-code or debugging information is available.

Publications

Qiushi Wu, Aditya Pakki, **Navid Emamdoost**, Stephen McCamant, and Kangjie Lu. "Understanding and Detecting Disordered Error Handling with Precise Function Pairing" *To appear in Proceedings of the 30th USENIX Security Symposium (Security'21). Vancouver, Canada, August 2021.*

Navid Emamdoost, Qiushi Wu, Kangjie Lu, and Stephen McCamant. "Detecting Kernel Memory Leaks in Specialized Modules with Ownership Reasoning" *in Proceedings of the 28th Network and Distributed System Security (NDSS) Symposium 2021.*

Mohsen Ahmadi, Pantea Kiaei, **Navid Emamdoost**. "SN4KE: Practical Mutation Testing at Binary Level" *In BAR 2021, NDSS Workshop on Binary Analysis Research. Internet Society, 2021.*

Vaibhav Sharma, **Navid Emamdoost**, Seonmo Kim, and Stephen McCamant. "It Doesn't Have to Be So Hard: Efficient Symbolic Reasoning for CRCs" *In BAR 2020, NDSS Workshop on Binary Analysis Research. Internet Society, 2020.*

Navid Emamdoost, Vaibhav Sharma, Taejoon Byun, and Stephen McCamant. "Binary Mutation Analysis of Tests Using Reassembleable Disassembly." *In BAR 2019, NDSS Workshop on Binary Analysis Research. Internet Society, 2019.*

Navid Emamdoost, Stephen McCamant. "The Effect of Instruction Padding on SFI Overhead." *In BAR 2018, NDSS Workshop on Binary Analysis Research. Internet Society, 2018.*

Navid Emamdoost, Mohammad Sadeq Dousti, and Rasool Jalili. "Statistical Disclosure: Improved, Extended, and Resisted." *In SECURWARE 2012, The Sixth International Conference on Emerging Security Information, Systems and Technologies, pp. 119-125. 2012.*

Navid Emamdoost, Mohammad Sadeq Dousti, and Rasool Jalili. "Improving Statistical Disclosure Attack on Anonymity Protocols." *In ISCISC 2010, The Seventh ISC International Conference on Information Security and Cryptology, pp. 33-40, 2010.*

Skills

Programming

- C/C++, C#, Rust, Python, JAVA, JavaScript, Assembly, WebAssembly

Database Management System

- PostgreSQL, MySQL

Tools

- Valgrind, LLVM, OProfile, DynInst, IDA Pro, Ollydbg, Git, Snort, Wireshark, Nmap, Tcpdump

Other

- Dynamic and Static Malware Analysis

Last updated on February 25, 2021