Mahmoud Abumandour

+ 1 (788) 320-8958 | mahmoud_mandour@sfu.ca | [Linkedin](#) | [GitHub](#)

EDUCATION

Simon Fraser University   **Sep 2022 – Aug 2024**
Master of Science in Computer Science (GPA: 4.0)   *BC, Canada*

Mansoura University   **Sep 2017 – Jul 2022**
Bachelor of Computer and Communication Engineering (GPA: 3.96, ranked first)   *Mansoura, Egypt*

PROFESSIONAL EXPERIENCE

Simon Fraser University   **BC, Canada**
Graduate Research Assistant   ***Sep 2022 – Present***

- Devised a fatal bit flip attack against quantized DNNs (degrades DenseNet121 into a random guesser with only 10 bit flips out of 58 million). Modified TensorFlow to use Triple Modular Redundancy and managed to reduce the attack success probability from 99% to 0.9% while incurring less than 1% performance overhead

Teaching Assistant   ***Jan 2023 – Present***

- Conducting tutorials and lab sessions, grading assignments and exams, and providing support during office hours
- Courses: Intro to computer systems, Principles of compiler design, Computer architecture

Google Summer of Code   **Remote**
Real-Time Executive for Multiprocessor Systems (RTEMS)   ***May 2022 – Sep 2022***

- Achieved 8x speedup over the previous release notes generator by using a multi-threaded architecture
- Automated manual Markdown and RST to PDF generation after fetching release data from RTEMS bug tracker

QEMU   ***May 2021 – Aug 2021***

- Implemented multi-core, multi-level cache performance emulation of user-space and full-system workloads
- Improved the system call tracing plugin by making its reports more script-friendly for post-processing

Master Micro   **Cairo, Egypt**
Software Engineering Intern   ***Oct 2021 – Feb 2022***

- Designed a database format for the main design lookup table file, reducing average query time by 50%
- Participated in code review and testing. Increased the testing coverage of the data querying subsystem by 10%

PROJECTS

- [**Fuzzing with RISC-V Emulation**](#): Developed a RISC-V 64-bit emulator suitable for userspace programs fuzzing, increasing test generation throughput by over 20x over single-core performance
- [**Database Engine**](#): Implemented a disk-oriented database management system with partial SQL-compliance, in-memory buffer pool caching, B+ Tree indexing, and JDBC driver
- [**AES Encryption Core**](#): Designed a low-power AES encryption core for FPGA. Reduced area and power consumption by 86% over a high-throughput pipelined design
- [**Hyperthreaded, Software-Interlocked MIPS Processor**](#): A multi-threaded five-stage pipelined MIPS core for FPGA and a custom interlocking-ensuring assembler, achieving 5x more throughput over single-threaded execution

SKILLS

Programming Languages: C++, C, x86 Assembly, Rust, Python, Bash Scripting, Java
Tools: Gem5, TensorFlow(lite/micro), git, Docker, Valgrind
Platforms: Linux, QEMU, FPGA, ARM Cortex M4, AVR
Hardware Design Tools: Xilinx Vivado, ModelSim, SystemVerilog, VHDL

OPEN-SOURCE CONTRIBUTIONS

- **SerenityOS**: Defined a global OS versioning API. Increased user-space utilities POSIX compliance. Improved the SerenityOS DBMS SQL compliance by supporting INSERTs with multiple tuples and table-description statements
- **QEMU**: Modernized the usage of locking and memory allocation APIs by using scope-based locks and automatically freed allocations. Redefined plugins' command line syntax adhering to modern QEMU CLI syntax