XIANMING LUO

984-377-9836 | xl369@duke.edu | LinkedIn | Personal Website

EDUCATION

Duke University, Pratt School of Engineering 08/2021 - 05/2023

- Major in Electrical and Computer Engineering (Software Engineering); GPA: **3.96/4.0**
- Key Courses: Operating Systems (**A+**), Advanced Computer Networks (**A**), Robust Server Software (**A**)

China University of Mining and Technology (CUMT) 09/2017 - 06/2021

- Major in Electronic and Information Engineering (Signal Processing); GPA: **91.53/100**; Rank: **7/394**
- Minor in Computer Science and Engineering; GPA: **92.64/100**; Rank:**1/56**

WORK EXPERIENCE

Oracle Exadata. Software Engineer Intern | Redwood City, CA 05/2022 - 08/2022

C/C++, Virtualization, RDMA, Networking, Concurrency, PMEM

- Emulated **Remote Direct Memory Access (RDMA)**-capable Network Interface Card (NIC) on host machines without RDMA-capable hardware using **Quick Emulator (QEMU)**. This allows developers to test RDMA code on emulated hardware instead of real hardware, which mitigated the real hardware shortage by **80%**
- Brought up nested **Kernel-based Virtual Machine (KVM)** with emulated RDMA-capable NICs on OCI VMs using Oracle Linux disk images in QEMU Copy On Write (QCOW2) format, **virsh**, and **virt-install**
- Set up **Virtual Networks** using **Linux Bridge** so that VMs can ping and do RDMA to each other
- Validated if the server software can correctly handle the poison in **Persistent Memory (PMEM)** by manually injecting poison to PMEM using Non-Volatile Device Control (ndctl)
- Parallelized the logical disk dropping process by applying asynchronous communication between services, accelerating the entire dropping process by roughly **5 times**

PROJECTS

Docker in xv6 (C, Kernel Programming, Container) 09/2022 - 12/2022

- Implemented PID Namespaces in xv6 with thorough testing, including reparenting, namespace collapsing, etc
- Evaluated the performance overhead of process running in docker and outside of docker

End-to-end Encrypted Multithreaded Chatroom (Python, TLS, Security) 09/2022 – 12/2022

- Developed an end-to-end encrypted chatroom, where the chatroom server can only see messages in ciphertext
- Established TLS connections between clients and the server using Python SSL
- Maintained a room master to distribute the chatroom's session key encrypted by other clients' public key

Mini UPS System (Java, Protocol Buffer, PostgreSQL, Multi-Thread) 04/2022 - 05/2022

- Using **Django**, developed a full-stack web application modeling UPS system, which simulated the whole process from truck request to truck arrival to incorporate with mini-Amazon developed by other groups
- Concurrently handled incoming truck requests and truck status updates using **Java Thread**, and applied asynchronous communication between mini-UPS and mini-Amazon using **Google Protocol Buffer**
- Implemented **idempotent** interfaces between mini-UPS and mini-Amazon and timeout-based retransmission mechanism to guarantee **Exactly-Once Semantics**

Strategic World Conquest Game (Java, Object-oriented Design, CI/CD, TCP Socket) 03/2022 - 04/2022

- Developed an online graphical game that players can attack territories, move soldiers, and chat with others
- Utilized **Object-oriented Design**, and applied **Design Patterns** to improve code maintainability
- Applied **Agile** methodology, and carried out issue tracking and **CI/CD** pipeline

LANGUAGE AND SKILLS

Language: C/C++, Java, Python, Bash, SQL, HTML, Swift, Verilog HDL

Tools: GDB, Linux, Git, Emacs, Make, ndctl, CI/CD, Google Protocol Buffer, Hibernate

Virtualization: libvirt, QEMU, KVM, virsh, virt-install