# Jiarui Li

jiaruil2@andrew.cmu.edu · (+1) 878-999-6130 · willendless

## Education

**Carnegie Mellon University**, Pittsburgh, PA August.2021 – May.2023(Expected)

*M.S.* in Information Networking, GPA 4.0/4.0

*Core courses* Storage Systems, Parallel Computer Architecture, Computer Graphics, Advanced Cloud Computing

**University of California, Berkeley**, CA, USA Jan.2020 – August.2020

*International Exchange Program* in EECS, GPA 3.97/4.0

*Core courses* OS and Systems Programming, Computer Security, Digital Design and Integrated Circuits

**East China Normal University**, Shanghai, China Sep.2017 – July.2021

*B.E.* in Software Engineering (Embedded System and Software concentration), GPA 3.81/4.0

## Experience

**PingCAP: TinyKV Study Camp** Golang, KV Storage System, Nov.2021 – Jan.2022

- Implemented a horizontally scalable, highly available multi raft-based kv distributed storage engine using Golang.
- Built a load-balancing scheduler to support dynamic configuration change and region split.
- Constructed a 2PC multi-version concurrency control layer to support distributed transaction.

## Projects

**SafeFS:** an End-to-End Encrypted File Sharing System Asymmetric/Symmetric Encryption, Golang

- Designed and built an encrypted file sharing client that allows users to securely store and share files to others.
- Used digital signature and hash with salt to secure user password and support authentication.
- Based on symmetric encryption technology, implemented encryption storage, file sharing and sharing revoke.

**RustOS:** a Rust AArch64 operating System in Raspberry Pi 3. AArch64, Rust

- Implemented a AArch64 operating system with virtual memory and multi-level queue scheduling support using Rust.
- Constructed a FAT32 file system for SD card and a LRU buffer cache to improve its performance.
- Developed GPU MailBox drivers and used framebuffer to enable font rendering to HDMI devices.

**CloudFS:** a hybrid cloud backed file system with snapshot support. FUSE, AWS S3, C++

- Constructed a hybrid user-level file system by storing small files in local and large files in Amazon S3 cloud storage.
- Used the content-based rabin fingerprinting to deduplicate large files to decrease the capacity cost. Also implemented snapshot to allow failure recovery and a ssd-based persistent cache to reduce S3 operations cost.

**Pintos:** a 80x86 time-sharing preemptive operating system QEMU, C

- Augmented the kernel with user program support by implementing arguments passing and process control syscalls. Also built a preemptive priority scheduler for the kernel.
- Developed a LRU buffer cache and a FFS like file system with extensible big files and sub directories support.

**MyFTL:** a page mapping log structured flash translation layer. SSD, C++

- Implemented a log structured flash translation layer to improve the endurance of the SSD.
- Used a greedy garbage collection policy to reduce write amplification and improve performance.
- Optimized the data structure and limited the memory cost to below 200kB and the write amplification to below 1.2.

**SuperNES:** a NES Emulator in Android. Emulator, Kotlin

- Emulated a 6502 CPU and 2C02 PPU with a dedicated render thread based on dynamic binary translation.
- Implemented a GUI game framework and ported 13 NES games into it, including Super Mario and Donkey Kong.
- Presented the app and NES hardware internals and received highest project scores (48/50) among all the teams.

## Skills

- Programming Languages: C/C++, Go, Java, Kotlin, Rust, SML, Python, Verilog
- Platforms: Linux, MAC, FPGA, Android, x86, AArch64