David H. Liu

Email: hao.liu@princeton.edu
Website: http://www.davidhliu.com
Phone: +1 (919) 667-6725
GitHub: @LedgeDash

EDUCATION

Princeton University, Princeton, NJ 2017 - 2023
Ph.D. in Computer Science
Thesis: *A Serverless Architecture for Application-Level Orchestration*
Advisors: Amit Levy, Michael Freedman

Duke University, Durham, NC 2011 - 2015
B.S.E. in Electrical and Computer Engineering
Minor in Mathematics

WORK EXPERIENCE

PhD in Computer Science 2017 - 2023
Princeton University, SNS Network Systems Group

- Projects and publications in serverless computing, light-weight virtualization, information-flow control, and Android security:
  - A novel orchestration system for large-scale serverless applications that ensures consistency and complex patterns distributedly entirely on the application-level. The system is compatible with a variety of distributed services including NoSQL databases, message queues, and object stores, and is portable across AWS and GCP.
  - A light-weight virtual machine that optimizes snapshot memoization and restoration based on the Firecracker VM by AWS Lambda. The system achieves up to 10x improvement in cold start latency compared with other state-of-the-art solutions.
  - A new Android operating system that performs dynamic information-flow tracking on both Java and native code. The system preserves parallelism and minimizes latency overhead of the added security measures by proposing a new approach that leverages ARMv7 Memory Domains to enforce per-thread memory page access privileges.

Research Intern Summer 2020
Microsoft Research, Mobility and Networking Group

- Built and profiled serverless systems and applications on the Azure Kubernetes Service (AKS)

Software Engineer 2015 - 2017
Nimble Storage, Data Protocol Team

- Developed firm's Linux device driver for the new Gen 6 Fibre Channel chipset
- Firm's liaison with Broadcom; led and tracked collaborative projects across companies

SKILLS

Python, Rust, JavaScript, Java, C, Kubernetes, PyTorch, Docker, SQL, MongoDB, Spark, AWS (Lambda, DynamoDB), GCP (Cloud Functions, Firestore), Azure (Azure Functions, Durable Functions), Linux kernel drivers

PUBLICATIONS

Doing More with Less: Orchestrating Serverless Applications without an Orchestrator
David H. Liu, Amit Levy, Shadi Noghabi, Sebastian Burckhardt
Proc. 20th Symposium on Networked Systems Design and Implementation (NSDI '23), Boston, MA, April 2023

How Low Can You Go? Practical cold-start performance limits in FaaS
Yue Tan, David H. Liu, Nanqinqin Li, Amit Levy

ArXiv Technical Report:2109.13319, Sept. 2021

Pyronia: Intra-Process Access Control for IoT Applications
Marcela S. Melara, David H. Liu, Michael J. Freedman
ArXiv Technical Report:1903.01950, March 2019

SandTrap: Tracking Information Flows On Demand with Parallel Permissions
Ali Razeen, David H. Liu, Alvin R. Lebeck, Alexander Meijer, Valentin Pistol, Landon P. Cox
The 16th ACM International Conference on Mobile Systems, Applications, and Services (MobiSys '18), June 2018

SELECTED PROJECTS

Unum
An orchestration system for large-scale serverless applications built on top of existing cloud services on AWS and Google Cloud Platform. Unum tackles many distributed systems challenges including consistency, exactly-once execution guarantees, and fault-tolerance. It improves programmability for complex serverless applications while significantly reducing latency and costs compared with existing orchestrators.

SnapFaaS
A light-weight virtual machine based on the Firecracker VM by AWS Lambda. SnapFaaS leverages a snapshotting technique to quickly restore VM states and reduce cold-start latency. SnapFaaS minimizes snapshot sizes and restoration latency by carefully identifying memory pages that are actually useful for application execution. SnapFaaS examines all stages of the VM boot process, including kernel loading, operating system init, language runtime setup, and application-specific initialization.

SandTrap
A dynamic information-flow tracking system on Android that performs native code taint tracking on the ARMv7 instruction set. SandTrap extends information flow control beyond JVM to native code by emulating ARMv7 instructions. SandTrap leverages memory domain to enforce per-thread memory page access privileges to preserve parallelism and minimize latency overhead.