Sun Guangda 孙广大 🔗

- Email: sung@comp.nus.edu.sg
- GitHub: sgdxbc (research) / whoiscc (personal)

Education 🔗

- Bachelor's degree from Xian JiaoTong University (2020)
- Ph.D student at National University of Singapore (2020-)

Interests 🔗

Research topics are building faulty tolerance systems for both data center and worldwide networks. Previous works also relate to software defined network, middleboxes and network virtualization.

Get used to: Rust, Python, C, Java, JavaScript, LaTex, asynchronous programming, network programming, Gurobi, Terraform, Vue, compiler frontend, memory allocator / garbage collection, Stack Overflow.
Learning: Coq, P4, Haskell, model checking, simulation, phsically based rendering, data processing, zero knowledge proof.

Played piano and listened to electronic music for a long time.

Works 🔗

NeoBFT 🔗

Byzantine fault tolerant (BFT) protocols are capable of masking sophisticated failures, but are rarely deployed due to their performance cost and complexity. In this work, we propose a new approach to designing high performance BFT protocols in data centers. By re-examining the ordering responsibility between the network and the BFT protocol, we advocate a new abstraction offered by the data center network infrastructure. Feasibility of the design is demonstrated by two hardware implementations - one using HMAC and the other using public key cryptography for authentication - on new-generation programmable switches. We then co-design a new BFT protocol, NeoBFT, that eliminates cross-replica coordination and authentication in the common case.

Guangda Sun, Mingliang Jiang, Xin Zhe Khooi, Yunfan Li, Jialin Li
NeoBFT: Accelerating Byzantine Fault Tolerance Using Authenticated In-Network Ordering pdf
SIGCOMM '23, doi: 10.1145/3603269.3604874

Repository: nus-sys/neobft-artifact

LemonNFV 🔗

A novel NFV framework that can consolidate heterogeneous NFs without code modification. LemonNFV loads NFs into a single process down to the binary level, schedules them using an intercepted I/O, and isolates them with the help of a restricted memory allocator.

Hao Li, Yihan Dang, **Guangda Sun**, Guyue Liu, Danfeng Shan, Peng Zhang
LemonNFV: Consolidating Heterogeneous Network Functions at Line Speed pdf
NSDI '23

Repository: ants-xjtu/neptune.

Rubik ⚭

Rubik is a network stack parser generator for network function middleboxes. It provides a programming model for network protocols and protocol stacks with handful abstractions like flow table and reordered buffer. With Rubik, user can write only hundreds of lines of configuration, which will be transformed into tens of thousands of lines of C code. The generated protocol parser is more efficient than any human-writing ones.

Hao Li, Changhao Wu, **Guangda Sun**, Peng Zhang, Danfeng Shan, Tian Pan, Chengchen Hu
Programming Network Stack for Middleboxes with Rubik pdf
NSDI '21

Repository: ants-xjtu/rubik.

CODER ⚭

To configure devices in a software defined network (SDN), e.g., packets routes, network functions location and resource allocation, serveral high-level abstractions and languages, like Merlin and SNAP, are developed. CODER provides an common immediate representation for all SDN languages, defines precise semantic of combining SDN programs, and lets SDN programs written in different languages work together. With CODER user has more power on configuring SDN.

Hao Li, Peng Zhang, **Guangda Sun**, Wanyue Cao, Chengchen Hu, Danfeng Shan, Tian Pan, Qiang Fu
Compiling Cross-Language Network Programs Into Hybrid Data Plane pdf
ToN '22, doi: 10.1109/TNET.2021.3132303.

Hao Li, Peng Zhang, **Guangda Sun**, Chengchen Hu, Danfeng Shan, Tian Pan, Qiang Fu
A Modular Compiler for Network Programming Languages pdf
CoNEXT '20, doi: 10.1145/3386367.3432063

Hao Li, Peng Zhang, **Guangda Sun**, Chengchen Hu, Danfeng Shan, Tian Pan, Qiang Fu
An Intermediate Representation for Network Programming Languages pdf
APNet '20, doi: 10.1145/3411029.3411030

Repository: ants-xjtu/sidfam-v3.

Projects ⚭

Shift Saver ⚭

- Repository: whoiscc/shift-saver
- Marketplace: Shift Saver
- Video