511, W Main St
Madison, WI 53703
kbavishi@cs.wisc.edu | karan.bavishi90@gmail.com
Webpage : https://kbavishi.github.io
Github : https://github.com/kbavishi
+1-608-504-0120

Karan Bavishi

Masters student, Computer Science, UW Madison

Education

University of Wisconsin-Madison, Madison, USA
Master of Science, Computer Science — *Aug '16 - May '18 (Expected)*
GPA: 4.0/4.0 (Overall)
Courses: Big Data Systems (CS838), Advanced Comp Architecture I (CS752)

Birla Institute of Technology and Science - Pilani, Pilani, India
Bachelor of Engineering (Hons.), Computer Science — *Aug '08 - May '12*
GPA: 8.08/10 (Overall)

Work Experience

ARISTA NETWORKS, Bangalore, India — *Jul '12 - Jul '16*
Software Engineer

License Management & High Availability

License management system to check that a switch is allowed to use the advanced software features present. Additional mechanisms to ensure high-availability via license state replication

- Helped design a distributed licensing system to replace the earlier system with no license enforcement.
- Integrated license awareness in features with guarantees of correct state regeneration across process restarts due to license enforcement.
- Added mechanisms using *inotify* and *rsync* to support directory-level replication of the license files across multiple nodes in the datacenter.
- Added fault tolerance mechanisms using clock timers and file locks to allow replication of directory snapshots. This ensured robustness to failures which could result in partial replication and thus cause state inconsistencies.

Audio Video Bridging

IEEE standards for delivering Audio/Video(A/V) traffic across an L2 network with guaranteed bandwidth, predictable low latency and correct time synchronization

- Worked on several areas of MSRP, to allow end-to-end resource reservation, misconfiguration detection, and rapid recovery support from link failures for A/V streams.
- Single-handedly implemented and tested MVRP, for dynamically provisioning VLANs in networks.
- Added L2 multicast support for the 7500E platform, to allow efficient bandwidth utilization.
- Improved scalability by 3× by adding support for batched updates to process state store. It helped avoid connection disruptions due to socket buffer overflows in situations involving heavy state churn.

IEEE 802.1X MAC Authentication Bypass (MAB)

MAB allows you to authenticate devices like printers that are not capable of IEEE 802.1X

- Added hardware support for trapping packets of unknown devices to CPU, so that the switch can authenticate them on their behalf. Also added rate-limiting to avoid overwhelming of CPU.
- Added hardware support for disabling MAC learning and not forwarding packets unless a static MAC entry was found. This allows forwarding of traffic from authenticated devices only.

Control-plane policing of SDN traffic

Policing for SDN traffic flowing between a controller and managed switches, to prevent throttling and starvation

- Added hardware rules in multiple switch platforms for trapping SDN traffic to the CPU, and classifying it to a traffic-class, to protect it from the aggressive shaping of unclassified traffic.
- Also added QoS rules for the SDN traffic to prevent starvation by other non-priority traffic. This reduced convergence time for services with hundreds of managed devices by nearly 40%.

Stateful Switchover (SSO) support for LLDP

Standby LLDP process running on the secondary supervisor to avoid failure downtime

- Added support for a standby LLDP process to run on the secondary supervisor to avoid downtime during failures and eliminate spurious SNMP traps about LLDP neighbors being down.

ARISTA NETWORKS, Bangalore, India *Jul '11 - Dec '11*
Software Intern

Userspace simulation of the kernelspace election module
Userspace simulation to allow the flexibility to write unit tests

- Implemented a userspace simulation of the kernel election module, responsible for deciding supervisor roles in a modular switch. This reduced testing time for various features by allowing unit tests.

Syslog rate-limiting and other improvements
Avoid log storage space exhaustion by rate-limiting syslogs based on passed arguments

- Added support for rate-limiting of syslogs by grouping and suppressing based on passed arguments.
- Discovered & fixed issues in the Python syslog infra, which improved performance by 50%.

Academic Projects

Performance analysis of NFs in different sandboxes
Supervisor : Prof. Aditya Akella *Sep '16 - Ongoing*

- Different sandbox environments such as Docker containers and Virtual Machines, offer different isolation levels. Stronger isolation can affect performance because of expensive context switches or other overheads. Weaker isolation can result in performance interference due to shared resources.
- Analyzed the impact of the overheads of different sandbox environments on the performance of NFs. Also analyzed the impact on performance due to CPU & Network contention.
- Report containing analysis of results to be available soon.

Improving performance of NFs by offloading to kernel
Supervisor : Prof. Aditya Akella *Dec '16 - Ongoing*

- Ongoing investigation to find potential of improving performance of NFs by offloading work to the kernel and thus avoid costly operations in userspace.

Storage-efficient solution for Geo-distributed Analytics
Supervisor : Prof. Aditya Akella | Course: Big Data Systems (CS838) *Oct '16 - Ongoing*

- Designed a storage-efficient solution for performing Geo-distributed Analytics by leveraging the use of Erasure Coding (EC) in HDFS. Achieved a reduction in WAN usage of about 2×.
- Added greedy EC read and write support to further improve performance.
- Further research being done related to developing a hybrid fault tolerance scheme.
- Github repo link. Course report link.

Design & Implementation of a 4-wide issue, out-of-order, superscalar processor
Supervisor : Prof. Karu Sankaralingam | Course : Adv Arch I (CS752) *Sep '16 - Dec '16*

- Designed a 4-wide issue, superscalar microprocessor from scratch, capable of running RV64I instructions in the RISCV ISA, with support for out-of-order execution, branch mispeculation recovery and partial ordering for memory disambiguation.
- Github repo link. Project report link.

Computer Skills

Languages (Expert): C, Python
Languages (Familar): C++, Java

References

1. Prof. Aditya Akella, *University of Wisconsin, Madison* Email: akella@cs.wisc.edu
2. VJ Shah, *Director, Arista Networks* Email: vshah@arista.com

Competitive Programming

Project Euler: Solved : 241/560 *(India Rank : 17)*