Andrew J. Gallatin

3910 Pumpkin Seed Ln
Glen Allen, VA 23060 USA

Voice: (804) 244-1876
Email: gallatin@gmail.com

Employment History

Netflix Inc   Los Gatos, CA (remote)
Senior Software Engineer, 2015-present
Supervised by Alex Gutarin and Jonathan Looney

I work for Netflix's Open Connect CDN kernel team. My focus is improving the efficiency of our CDN servers. In this role I:

- Led efforts to improve video serving performance by 10x, from less than 40Gb/s per server 2015 to nearly 400Gb/s in 2021.
- Designed and implemented the FreeBSD Kernel TLS (kTLS) subsystem for Netflix, which was later upstreamed into FreeBSD.
- Designed a new FreeBSD network buffer (mbuf) type to improve efficiency for kTLS and sendfile.
- Designed and implemented a mechanism to maintain NUMA connection locality that roughly doubled video serving performance on NUMA hosts.
- Worked on the initial implementation of several FreeBSD NUMA optimizations, such as backing the page tables with NUMA-local memory.
- Designed and co-implemented a shortcut for firewall software to examine ingress traffic via a device driver upcall. This more than doubled the amount of denial of service traffic a server could withstand while still serving video traffic.
- Identified and fixed numerous scalability bottlenecks in the FreeBSD kernel using profiling tools (Vtune, AMDuProf, hwpmc) and observability tools (Dtrace).
- Worked closely with the Netflix Open Connect hardware team to bring up and evaluate new platforms.
- Ported our CDN server software stack from FreeBSD/amd64 to FreeBSD/arm64.
- Worked with numerous external hardware vendors to improve the performance of their hardware on FreeBSD.

Google Inc   Mountain View, CA
Staff Software Engineer, 2015
Senior Software Engineer, 2013-2015
Supervised by Jakov Seizovic

I worked for Google's platforms and networking group on device drivers and firmware for in-house silicon. In this role I:

- Contributed to the specification and design of in house silicon, from a driver interface perspective.
- Worked across teams to gather requirements and input into in-house silicon features.

- Wrote and optimized device drivers for the Google production Linux kernel.
- As my 20% project, I more than doubled the efficiency of the Akaros kernel's Plan 9 based network stack by implementing modern features, such as checksum offload, TCP Large Send, and TCP Large Receive Offload.
- Was promoted from Senior to Staff Software Engineer (L5 to L6).

Myricom Inc   Arcadia, CA (remote)
Software Product Architect, 2010-2013
Member of the Technical Staff, 2001-2010

Supervised by Jakov Seizovic

I was the leader of Myricom's 10 Gigabit ethernet device driver department. I was the sole maintainer of Myricom's Linux and Unix device drivers, and I managed a Windows developer who works on the Windows drivers. In this role, I:

- Participated in the specification and design of a cutting edge next-generation 10GbE NIC. I worked on efficient transmit and receive descriptors, and an innovative approach to interrupt handling.
- Assisted the hardware team with high-level verification of the 10GbE NIC using verilog software and hardware simulators in conjunction with virtual and physical machines running our device drivers.
- Wrote and maintained high performance network device drivers on Linux, FreeBSD, VMware ESX, Solaris, and Mac OS X, AIX, and DEC UNIX NICs for multiple generations of NICs.
- Worked with multiple OS vendors to get our device drivers included in their base installation.
- Collaborated with outside developers on Myri10GE drivers for their platforms (Plan9, NetBSD, OpenBSD, etc).
- Was part of a small team which designed and implemented Myricom's Myrinet Express *(MX)* OS-bypass software platform. Products such as DBL, Sniffer, and VideoPump are all derived from MX.

Department of Computer Science, Duke University   Durham, NC
Senior Systems Programmer 1996-2001

Supervised by Prof. Jeffrey Chase

- Helped port GMS, a kernel based distributed memory system, from DEC Unix to FreeBSD.
- One of a small team which designed and implemented Trapeze, our own messaging system for Myricom's Lanai4 NICs.
- Implemented zero-copy socket extensions in DEC Unix and FreeBSD.
- Reduced overhead and increased bandwidth of FreeBSD's NFS client code by eliminating double buffering, extra copies, and frequent NFSv3 commits.
- Ported the IProbe Alpha profiling tool to FreeBSD/alpha.
- Ported drivers and support code for various devices to FreeBSD, incliding Myricom NICs, Giganet Clan VIA drivers, Compaq's PCI Pamette.

- Ported FreeBSD from 32-bit x86 to the 64-bit DEC Alpha, as a member of a two person team.
- Fixed many 32/64 bit bugs in the kernel and in userspace applications.
- Ported NetBSD's DEC Unix binary compatibility layer to FreeBSD/alpha and extended it to enable FreeBSD/alpha to run dynamically linked commercial DEC Unix executables, such as Mathematica.
- Implemented the Linux/alpha binary compatibility layer in FreeBSD/alpha.

INSTITUTE OF STATISTICS AND DECISION SCIENCES, Duke University   Durham, NC
Computer Project Manager 1993-1996

- Performed Unix systems administration for a network of roughly 30 DEC Unix and DEC Ultrix workstations.
- Ported 32-bit open-source applications to 64-bit DEC Unix.

Education

RENSSELAER POLYTECHNIC INSTITUTE
M.S. in Computer Science, 1993
GPA: 4.0/4.0

STATE UNIVERSITY OF NEW YORK AT BUFFALO
B.S. in Computer Science, 1992
GPA: 3.95/4.0 (Summa Cum Laude)

Publications

GALLATIN, D. NUMA Optimizations in the Freebsd Network Stack. https://papers.freebsd.org/2019/eurobsdcon/gallatin-numa_optimizations_network_stack, Sep 2019

SHWARTSMAN, S., AND GALLATIN, D. Kernel TLS and TLS hardware offload. https://papers.freebsd.org/2019/eurobsdcon/shwartsman_gallatin-kernel_tls_harware_offload, Sep 2019

GALLATIN, D. Serving 100 Gbps from an Open Connect Appliance. https://netflixtechblog.com/serving-100-gbps-from-an-open-connect-appliance-cdb51dda3b99, Sep 2017

MAGOUTIS, K., ADDETIA, S., FEDOROVA, A., SELTZER, M., CHASE, J., GALLATIN, A., KISLEY, R., WICKREMESINGHE, R., AND GABBER, E. Structure and Performance of the Direct Access File System. In *Proceedings of the 2002 USENIX Technical Conference* (2002)

CHASE, J., GALLATIN, A., AND YOCUM, K. End System Optimizations for High-Speed TCP. *Communications Magazine, IEEE 39*, 4 (2001), 68–74

CHASE, J., ANDERSON, D., GALLATIN, A., LEBECK, A., AND YOCUM, K. Network I/O with Trapeze. In *1999 Hot Interconnects Symposium* (1999)

GALLATIN, A., CHASE, J., AND YOCUM, K. Trapeze/IP: TCP/IP at Near-Gigabit Speeds. In *1999 USENIX Technical Conference (Freenix Track)* (1999)

YOCUM, K., ANDERSON, D., CHASE, J., GADDE, S., GALLATIN, A., LEBECK, A., ET AL. Adaptive Message Pipelining for Network Memory and Network Storage. In *Eighth International Conference on Architectural Support for Programming Languages and Operating Systems* (1998)

ANDERSON, D., CHASE, J., GADDE, S., GALLATIN, A., YOCUM, K., AND FEELEY, M. Cheating the I/O bottleneck: Network Storage with Trapeze/Myrinet. In *Proceedings of the 1998 USENIX Technical Conference* (1998), vol. 182

YOCUM, K., CHASE, J., GALLATIN, A., AND LEBECK, A. Cut-through delivery in Trapeze: An Exercise in Low-latency Messaging. In *High Performance Distributed Computing, 1997. Proceedings. The Sixth IEEE International Symposium on* (1997), IEEE, pp. 243–252