L41: Readings

Dr Robert N.M. Watson

Michaelmas Term 2016

Reading assignments

Reading assignments should be completed prior to the lecture or lab that they correspond to. Full citations for books and papers may be found below.

Lecture 1: McKusick, et al: Chapter 2 (*Design Overview of FreeBSD*).

Lecture 2: McKusick, et al: Chapter 3 (*Kernel Subsystems*).

Paper - Cantrill, et al. 2004.

Lab 1: Gregg and Mauro: Chapters 1 (*Introduction to DTrace*) and 2 (*D Language*).

Lecture 3: McKusick, et al: Chapter 4 (*Process Management*).

Paper - Anderson, et al. 1992.

Lecture 4: McKusick, et al: Chapter 6 (*Memory Management*).

Paper - Navarro, et al. 2002.

Lab 2: Paper - Ellard and Seltzer 2003.

Lab 3: *No reading assignment*.

Lecture 5: McKusick, et al: Chapter 12 (*Inter-Process Communication*).

Paper - Rizzo 2012.

Lecture 6: McKusick, et al: Chapter 14 (*Transport-Layer Protocols*).

Paper - Marinos, et al. 2014.

Lab 4: Paper - Bishop, et al. 2005.

Lab 5: Paper - Van Jacobson 1988.

Course texts

Course texts provide instruction on statistics, operating-system design and implementation, and system tracing. You will be asked to read selected chapters from these, but will likely find other content in them useful as you proceed with the labs.

Marshall Kirk McKusick, George V. Neville-Neil, and Robert N. M. Watson. *The Design and Implementation of the FreeBSD Operating System, 2nd Edition*, Pearson Education, Boston, MA, USA, September 2014.

Brendan Gregg and Jim Mauro. *DTrace: Dynamic Tracing in Oracle Solaris, Mac OS X and FreeBSD*, Prentice Hall Press, Upper Saddle River, NJ, USA, April 2011.

Raj Jain, *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*, Wiley - Interscience, New York, NY, USA, April 1991.

Research readings

Our research readings are drawn from various systems publications venues; these provide insight into types of research done with systems that are particularly relevant to our laboratory work, but also examples of practical systems research. Some readings are assigned prior to specific lectures or labs; others are for your (optional) enlightenment (and hopefully also enjoyment).

Tracing and performance analysis

Bryan M. Cantrill, Michael W. Shapiro and Adam H. Leventhal. *Dynamic Instrumentation of Production Systems*. Proceedings of the 2004 USENIX Annual Technical Conference, USENIX Association, June, 2004.

Daniel Ellard and Margo Seltzer. *NFS Tricks and Benchmarking Traps*. Proceedings of the 2003 USENIX Annual Technical Conference, FREENIX Track, USENIX Association, June, 2003.

Luigi Rizzo. *Dummynet: A Simple Approach to the Evaluation of Network Protocols*, ACM SIGCOMM Computer Communication Review 27(1), 31-41, ACM, 1997. **(Optional reading)**

Kernel structure and primitives

Mike Accetta, Robert Baron, William Bolosky, David Golub, Richard Rashid, Avadis Tevanian, and Michael Young. *Mach: A New Kernel Foundation for UNIX Development*. Proceedings of the 1986 USENIX Summer Conference, USENIX Association, June, 1986. **(Optional reading)**

Thomas E. Anderson, Brian N. Bershad, Edward D. Lazowska, and Henry M. Levy. *Scheduler Activations: Effective Kernel Support for User-Level Management of Parallelism*. ACM Transactions on Computer Systems, 10(1), 53-79, ACM, February 1992.

Juan Navarro, Sitaram Iyer, Peter Druschel, Alan L. Cox. *Practical, Transparent Operating System Support for Superpages*. 5th Symposium on Operating Systems Design and Implementation (OSDI '02), USENIX Association, December, 2002.

Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt, and Andrew Warfield. *Xen and the Art of Virtualization*. Proceedings of the 19th ACM Symposium on Operating Systems Principles (SOSP'03), ACM, October 2003. **(Optional reading)**

Silas Boyd-Wickizer, Austin T. Clements, Yandong Mao, Aleksey Pesterev, M. Frans Kaashoek, Robert Morris, and Nickolai Zeldovich. *An Analysis of Linux Scalability to Many Cores*. Proceedings of the 9th USENIX Symposium on Operating System Design and Implementation (OSDI '10), USENIX Association, October, 2010. **(Optional reading)**

Network stacks

Steve Bishop, Matthew Fairbairn, Michael Norrish, Peter Sewell, Michael Smith, and Keith Wansbrough. *Rigorous Specification and Conformance Testing Techniques for Network Protocols, as Applied to TCP, UDP, and Sockets*. Proceedings of SIGCOMM 2005, ACM, 2005.

Van Jacobson. *Congestion avoidance and control*. Proceedings of SIGCOMM, ACM, 1988.

Steven McCanne and Van Jacobson. *The BSD Packet Filter: A New Architecture for User-level Packet Capture*. Proceedings of the 1993 USENIX Winter Conference, USENIX Association, January, 1993. **(Optional reading)**

Luigi Rizzo. *netmap: A Novel Framework for Fast Packet I/O*. Proceedings of the USENIX 2012 Annual Technical Conference (ATC'12), USENIX Association, June, 2012.

Ilias Marinos, Robert N. M. Watson, and Mark Handley, *Network Stack Specialization for Performance*, Proceedings of SIGCOMM, ACM, August, 2014.

Supplemental course texts

The supplemental readings may be useful in refreshing or building up your basic knowledge and skills in support of our lectures and labs.

Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne, *Operating System Concepts, Eighth Edition*, John Wiley and Sons, Inc., New York, NY, USA, July 2008.

Brendan Gregg. *Systems Performance: Enterprise and the Cloud*, Prentice Hall Press, Upper Saddle River, NJ, USA, October 2013.

Wes McKinney, *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*, O'Reilly Media, Sebastopol, CA, USA, October 2012.

Websites

These websites may also be of use:

L41 Module Page	`https://www.cl.cam.ac.uk/teaching/1617/L41/`
FreeBSD Project	`https://www.FreeBSD.org/`
FreeBSD Subversion Repository	`https://svn.FreeBSD.org/`
DTrace on FreeBSD	`https://wiki.freebsd.org/DTrace`
FreeBSD and Linux Kernel Cross-Reference	`http://fxr.watson.org/`
FreeBSD Benchmark Advice	`https://wiki.freebsd.org/BenchmarkAdvice`
BeagleBone Black	`http://beagleboard.org/black`