

CSC 469H1 F / CSC 2208H1 F

ADVANCED OPERATING SYSTEMS

Fall 2007

Note: Some of the readings listed here will be “recommended”, not “required”; additional readings may be added as the course progresses. For the definitive list, follow the course website.

List of Topics

1. Operating system design and internal structure [1, 2, 3, 4, 5, 6, 7]
 - monolithic kernels, microkernels and open systems
 - OS extensions (kernel modules, user-level servers, sandboxing)
 - Virtual machines
2. Performance evaluation and benchmarking [8]
 - Measures of time, interval counting and cycle counting
 - Measurement pitfalls – overhead and perturbation
 - Using measurements – Amdahl’s Law
3. Inter-process communication (IPC) [9]
 - Interrupt handling
 - Signal handling
 - Implementation of sockets and pipes
 - Event notification (poll, select, epoll and kqueue) – tradeoffs and implementations
4. Multiprocessor synchronization [10, 11]
 - Dealing with relaxed memory consistency models
 - Spinlock alternatives for reduced contention and improved scalability
 - Avoiding locks – the Linux Read-Copy-Update (RCU) API
 - Transactional Memory
5. Multiprocessor scheduling [12, 13]
 - Scheduling considerations - design of ready queue, load balancing and processor affinity
 - Parallel job scheduling
6. Virtual memory [14, 15, 16, 17, 18]
 - Page table and TLB design for large address spaces
 - Superpages
 - Page placement for better cache behavior, reduced power, or improved NUMA performance
 - Software distributed shared memory
7. Distributed systems [19, 20, 21]
 - Time, clocks, and event ordering
 - Distributed agreement protocols
 - Fault tolerance and replicated state machines
8. Reliable, high performance storage [22, 23]

- Log-structured file systems
 - Soft updates
 - RAID
9. Security [24, 25]
- Threats, vulnerabilities and attacks
 - OS security mechanisms – access controls, FreeBSD Jails and SELinux

Reading List

- [1] J. H. Saltzer, D. P. Reed, and D. D. Clark. End-to-end arguments in system design. *ACM Transactions on Computer Systems*, 2(4):277–288, 1984.
- [2] Butler Lampson. Hints for computer system design. *ACM Operating Systems Rev*, 15(5):33–48, 1983.
- [3] Dennis M. Ritchie and Ken Thompson. The UNIX time-sharing system. *Communications of the ACM*, 17(7):365–375, 1974.
- [4] Michael J. Accetta, Robert V. Baron, William J. Bolosky, David B. Golub, Richard F. Rashid, Avadis Tevanian, and Michael Young. Mach: A new kernel foundation for unix development. In *USENIX Summer*, pages 93–113, 1986.
- [5] Brian N. Bershad, Stefan Savage, Przemyslaw Pardyak, Emin Gün Sirer, Marc E. Fiuczynski, David Becker, Craig Chambers, and Susan J. Eggers. Extensibility, safety and performance in the SPIN operating system. In *15th Symposium on Operating Systems Principles*, pages 267–284, 1995.
- [6] Greg Kroah-Hartman. Signed kernel modules. *Linux Journal*, (117), January 2004.
- [7] Samuel T. King, George W. Dunlap, and Peter M. Chen. Operating system support for virtual machines. In *USENIX Annual Technical Conference, General Track*, pages 71–84, 2003.
- [8] Carl Staelin. *lmbench*: an extensible micro-benchmark suite. *Software, Practice and Experience*, 35(11):1079–1105, 2005.
- [9] Jonathan Lemon. Kqueue - a generic and scalable event notification facility. In *Proceedings of the FREENIX Track: 2001 USENIX Annual Technical Conference*, pages 141–153, 2001.
- [10] John M. Mellor-Crummey and Michael L. Scott. Algorithms for scalable synchronization on shared-memory multiprocessors. *ACM Transactions on Computer Systems*, 9(1):21–65, 1991.
- [11] Andrea Arcangeli, Mingming Cao, Paul E. McKenney, and Dipankar Sarma. Using read-copy-update techniques for system v ipc in the linux 2.5 kernel. In *USENIX Annual Technical Conference, FREENIX Track*, pages 297–309, 2003.
- [12] Dror G. Feitelson and Larry Rudolph. Parallel job scheduling: Issues and approaches. In *IPPS '95: Proceedings of the Workshop on Job Scheduling Strategies for Parallel Processing*, pages 1–18, 1995.
- [13] Dror G. Feitelson, Larry Rudolph, and Uwe Schwiegelshohn. Parallel job scheduling - a status report. In *Proceedings of the Tenth Workshop on Job Scheduling Strategies for Parallel Processing*, pages 1–16, 2004.
- [14] Madhusudhan Talluri, Mark D. Hill, and Yousef A. Khalidi. A new page table for 64-bit address spaces. In *Proceedings of the fifteenth ACM Symposium on Operating Systems Principles*, pages 184–200, 1995.
- [15] Juan Navarro, Sitaram Iyer, Peter Druschel, and Alan Cox. Practical, transparent operating system support for superpages. In *Proceedings of the 5th Symposium on Operating Systems Design and Implementation*, pages 80–104, 2002.
- [16] R. E. Kessler and Mark D. Hill. Page placement algorithms for large real-indexed caches. *ACM Transactions on Computer Systems*, 10(4):338–359, 1992.

- [17] Bill Nitzberg and Virginia Lo. Distributed shared memory: A survey of issues and algorithms. *IEEE Computer*, 24(8):52–60, 1991.
- [18] Cristiana Amza, Alan L. Cox, Sandhya Dwarkadas, Pete Keleher, Honghui Lu, Ramakrishnan Rajamony, Weimin Yu, and Willy Zwaenepoel. Treadmarks: Shared memory computing on networks of workstations. *IEEE Computer*, 29(2):18–28, 1996.
- [19] Leslie Lamport. Time, clocks, and the ordering of events in a distributed system. *Communications of the ACM*, 21(7):558–565, 1978.
- [20] Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, 1982.
- [21] Fred B. Schneider. Implementing fault-tolerant services using the state machine approach: a tutorial. *ACM Computing Surveys*, 22(4):299–319, 1990.
- [22] Mendel Rosenblum and John K. Ousterhout. The design and implementation of a log-structured file system. In *Proceedings of the thirteenth ACM Symposium on Operating Systems Principles*, pages 1–15, 1991.
- [23] Gregory R. Ganger, Marshall Kirk McKusick, Craig A. N. Soules, and Yale N. Patt. Soft updates: a solution to the metadata update problem in file systems. *ACM Transactions on Computer Systems*, 18(2):127–153, 2000.
- [24] Eugene H. Spafford. Crisis and aftermath. *Communications of the ACM*, 32(6):678–687, 1989.
- [25] Ray Spencer, Stephen Smalley, Peter Loscocco, Mike Hibler, Dave Andersen, and Jay Lepreau. The flask security architecture: System support for diverse security policies. In *Proceedings of the 8th USENIX Security Symposium*, page 18 pages, 1999.