

## **COSC 6360: Operating Systems**

### **MATERIALS ON THE SECOND SUMMER 2008 MIDTERM**

You are responsible for the materials discussed in class as they are *summarized* in the handouts and/or mentioned in the PowerPoint presentations. I expect you to understand these summaries and to be able to comment around them.

#### **Clock Synchronization**

L. Lamport, "Time, clocks and the ordering of events in a distributed system," *CACM*, 21(7) 558-565, July 1978.

- *We covered most aspects of the paper: skip the proof at the end.*

#### **Interprocess Communication**

C. A. R. Hoare, "Communicating Sequential Processes," *CACM*, 21(8):666-677, Aug. 1978.

- *Focus on the discussion of guarded commands and alternative commands. You must be able to read CSP but will never be asked to write CSP code.*

L. E. Moser, P. M. Melliar-Smith, D. A. Agarwal, R. K. Budhia, and C. A. Lingley-Papadopoulos, "Totem: a fault-tolerant multicast group communication system," *CACM*, 39(4) April 1996.

- *We covered this paper in some detail. You should understand the two reliable totally ordered message delivery services that Totem provides and their implementation; how the single ring protocol orders messages and the role of guaranteed vector messages.*

#### **Security**

B. W. Lampson, "A note on the confinement problem," *CACM*, 16(10):613-615, Oct. 1973.

- *We did not cover the paper.*

J. G. Steiner, C. Neuman and J. I. Schiller, "Kerberos: An authentication service for open network systems," *Proc. 1988 Winter USENIX Conference*, pp. 205-211, Feb 1988.

- *We covered the paper in some detail. You should understand all the materials in the slides.*

#### **Secondary Storage Organizations**

D. A. Patterson, G. A. Gibson, and R. H. Katz. "A case for redundant arrays of inexpensive disks (RAID)," *Proc. SIGMOD International Conference*, pp. 109-116, June 1988.

- *You should understand the general motivation for RAID organizations and the small write problem in RAID 5. Skip the availability analysis: it is wrong.*

#### **File Systems**

M. Rosenblum and J. K. Ousterhout, "The design and implementation of a log-structured file system," *ACM Transactions on Computer Systems*, 10, 1 (Feb. 1992), 26-52.

- *You should understand the motivations for a log-structured file system, its general organization (including the new i-node map) and the trade-offs to be considered in the segment cleaning process. (LFS-3.ppt)*

M. Seltzer, K. Bostic, M. K. McKusick, C. Staelin, "A log-structured file system for UNIX," *Proc. 1993 Winter Usenix Conference*, (1993).

- *We did not cover the paper in detail and focused on how BSD-LFS addresses some of the limitations of FFS, among which, the lack of a comprehensive file system check at start time (BSD-LFS runs fschk in the background). You should also be aware that disk controllers can and sometimes do reorder write requests. (BSD-LFS slides 1-17) much*

M. I. Seltzer, G. R. Ganger, M. K. McKusick, K. A. Smith, C. A. N. Soules, and C. A. Stein, “Journaling versus soft updates: Asynchronous meta-data protection in file systems,” *Proceedings of the 2000 USENIX Annual Technical Conference*, June 2000, pages 71-84.

- *You should understand soft updates and journaling in some detail. You should also know—and understand--the major conclusions of the paper.*

## Distributed File Systems

R. Sandberg, D. Goldberg, S. Kleiman, Dan Walsh and Bob Lyon, “Design and implementation of the Sun network filesystem,” *Proc. Summer 1985 USENIX Conference*, pp. 119-130.

- *We covered the paper in some detail. You should understand why the designers of NFS decided to select a stateless server and how their decision affected the overall design of the system. Do not go at a much lower level of detail than the PowerPoint slides. (NFS.ppt)*

D. Hitz and A. Watson, “The Evolution of NFS,” Network Appliance, Inc.  
[http://www.netapp.com/tech\\_library/evolution.html](http://www.netapp.com/tech_library/evolution.html)

- *You are responsible for all the topics discussed in the PowerPoint slides (NFS Evolution.ppt)*

## Selected Revision Questions

1. A system of physical clocks consists of two clocks, namely, one that is slow and loses 5 minutes every hour and another that is fast and advances by 5 minutes every hour. Assuming that the clocks are managed by Lamport’s physical clock protocol, what will be the time marked by each clock at 2:00 PM given that (a) both clocks indicated the correct time at noon; (b) the processors on which the clock reside continuously exchange messages between themselves; and (c) the message transmission delays are negligible.
2. Why do neither C nor C++ include any statements comparable to the CSP guarded command and alternative command?
3. What is the function of Totem *guaranteed vector messages*?
4. A Totem system has three rings A, B and C. Which messages will a processor X be able to deliver using *agreed delivery* assuming that it has received but not yet delivered messages with the following timestamps?

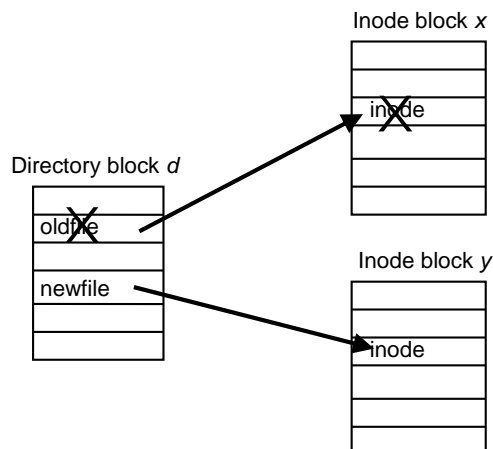
From ring A: 4:50 PM

From ring B: 4:55 PM and 5:00 PM

From ring C: 4:45 PM and 5:05 PM,

5. Give three examples of *covert channels* a malicious process can use to pass information.
6. What would be the consequences of having an intruder penetrate (a) the Kerberos server and (b) the Kerberos Ticket Granting Service?
7. What is the purpose of Kerberos authenticators?
8. A RAID level 5 array consists of  $n$  drives. What is the most efficient way to update a single block whose previous contents were never read into memory when (a)  $n = 3$  and (b)  $n = 8$ ?
9. What is the purpose of the *i-node map* of a log-structured file system?

10. Consider a file system using soft updates and assume that one directory block in its I/O buffer reflects the result of one file deletion and one file creation. Assuming that the i-nodes of the two files reside in two different i-node blocks, describe each step the system will take to update the disk copies of the three blocks. (3×5 points) (*Hint: treat this question as a problem.*)



11. What are the main advantage and the main disadvantage of *stateless servers*?
12. What is a *stale file handle*? What does NFS do to detect them?
13. Consider a *single-ring* Totem system comprising two processors A and B. Assuming that each of these two processors has received the messages with the following sequence numbers:

| Processor | Messages      |
|-----------|---------------|
| A         | 3, 4, 6, 8    |
| B         | 3, 4, 5, 7, 8 |

Which messages will be delivered by each processor if all messages are *safe delivery messages*?