Name: _____, Student ID: _____

CSE 536 Spring 2026 Exam 1

Assignment ID: 0

Instructions

Select the one best answer. Don't forget to write your name and student ID at the top. **You must show your ID as you turn in your exam.** You have an hour and 15 minutes (a regular class period) to complete this exam (unless you are taking it in another location with different arrangements). Mark on these sheets of paper with a pen or pencil, and then turn it in at the front of the room to the TA or I. You may not use scratch paper or notes of any kind, so there should be no other pieces of paper on your desk during the exam and you should not write on anything other than the exam itself. This exam is closed book (note that there is no textbook for the course) and closed note. You may use a calculator, but only a Casio fx-115ES PLUS (you won't need it for CSE 536 Exam 1). You may not use any other electronic device (and especially not a cell phone). You may not communicate in any way with any individuals other than the instructor of the course, the TA, or another official proctor during the exam. Any violation of these policies will result in a 0 on the exam and will be treated as an act of academic dishonesty as per the syllabus. The exam is worth 100 points total. This is printed exam number: 0 (you can safely ignore this number).

1. Why did the simple solution of simply dropping privileges to open a file not used to be a satisfactory solution to the problem of TOCTTOU race conditions where setuid binaries access files on a user's behalf?

A Dropping privileges is not reversible
B File descriptors were not a standard UNIX concept until very recently
C The semantics of the setuid() family of systems calls varied widely across UNIX implementations
D UNIX systems typically don't have preemptive schedulers

2. Suppose a process opens a file, and then reads from it. When are the permissions in the file's inode checked to make sure the process is allowed to read from the file?

A When the dirty page is synched to the block device
B Never, because UNIX has no file protection mechanisms
C When the file is read
D When the file is opened

3. What did Leslie Lamport recommend for ensuring the correctness of a distributed system?

A Thinking in terms of partial orderings of events
B Using perfectly synchronized atomic clocks to mark all events
C Always using counting semaphores, never binary semaphores
D Completely removing any notion of "processes" from the system

4. Linux's Completely Fair Scheduler and Solaris's Multilevel Feedback Queue are examples of what principle used for scheduling reactive systems that interact with a user?

A Counting semaphores
B Non-preemptive kernel scheduling
C Giving a higher dynamic priority to I/O intensive processes
D Monitors

5. Which of these is <u>NOT</u> a feature of the mutexes that the Linux kernel uses for avoiding race conditions?

A It is reentrant
B A multi-path implementation with a spinlock and a wait queue
C Priotity inheritance
D A binary semaphore

6. Notwithstanding ROWHAMMER or other bugs/vulnerabilities, when is it considered safe for the kernel to map a file owned by root into a process's address space, where the user who owns the process has read permissions on the file, but not write permissions.

A Never

B When it's a read-only mapping or Copy-on-Write private mapping

C Only when the process is owned by the root user

D Only when the process is a bash shell

7. In the access()/open() TOCTTOU filesystem race condition that we dicsussed in class, what is the victim setuid root process typically doing that might get interrupted, leading to the race condition?

A Writing to a named pipe

B Traversing d_entry's and inodes to complete the access() and open() system calls

C Handling a SIGINT signal

D Writing to an unnamed pipe

8. Which of these would a process to do create the highest ratio of physical page frames dedicated to page tables *vs.* **physical page frames mapped into a process's address space?**

A malloc() a lot of small arrays and never access them

B Map a very large file the size of available physical memory

C Map the same small file over and over throughout its virtual address space, and access every part of each mapping

D malloc() a huge array and then sequentially writes to the whole thing

9. Under which one of the following circumstances might a process find itself in the "pipe wait" state, waiting for I/O before it can be scheduled on the CPU again?

A When it tries to access protected memory and then doesn't handle the SIGSEGV signal

B When another process is hogging the CPU

C When it reads from a pipe

D When the machine is turned off

10. If I use a browser on my Linux laptop in incognito mode, so that my web activity is not stored to any file, how long might information about which network hosts I communicated with persist on that laptop?

A Only until the entry is deallocated from the slab allocator

B Only until the entry is deallocated from the buddy heap

C Only until the next time the laptop is powered off

D Potentially for years, because of virtual memory swapping

This page intentionally left blank.