Homework #4

Due: May 27, 2026 **Points:** 100

Questions

1. (*20 points*) Suppose a user wishes to edit the file *xyzzy* in a capability-based system.
   (a) How should he do this to ensure that the editor cannot access any other file?
   (b) Could this be done in an ACL-based system? If so, how? If not, why not?

2. (*24 points*) Consider Multics procedures p and q. Procedure p is executing and needs to invoke procedure q. Procedure q's access bracket is (5, 6) and its call bracket is (6, 9). Assume that q's access control list gives p full (read, write, append, and execute) rights to q. In which ring(s) must p execute for the following to happen?
   (a) p can invoke q, but a ring-crossing fault occurs.
   (b) p can invoke q provided that a valid gate is used as an entry point.
   (c) p cannot invoke q.
   (d) p can invoke q without any ring-crossing fault occurring, but not necessarily through a valid gate.

3. (*28 points*) Classify the following vulnerabilities using the PA model. Assume that the classification is for the implementation level. Justify your answer.
   (a) The presence of the "wiz" command in the *sendmail* program (see Section 24.2.9).
   (b) The failure to handle the `IFS` shell variable by *loadmodule* (see Section 24.2.9).
   (c) The failure to select an *Administrator* password that was difficult to guess (see Section 24.2.10).
   (d) The failure of the Burroughs system to detect offline changes to files (see Section 24.2.7).

4. (*28 points*) StackGuard is a tool for detecting buffer overflows. It modifies the compiler to place a known (pseudo)random number (a *canary*) on the stack just before the return address when a function is called. Additional code is added so that, just before the function returns, it pops the canary and compares it to the value that was placed upon the stack. If the two differ, StackGuard asserts a buffer overflow has occurred, and invokes an error handler to terminate the program. How effective is this approach at stopping stack-based buffer overflows? Under what conditions might it fail?

Extra Credit

E1. (*15 points*) Discuss controls that would prevent Dennis Ritchie's bacterium from absorbing all system resources and causing a system crash.

E2. (*20 points*) A vendor advertises that its system was connected to the Internet for 3 months, and no one was able to break into it. It claims that this means the system cannot be broken into from any network.
   (a) Do you share the vendor's confidence? Why or why not?
   (b) If a commercial evaluation service had monitored the testing of this system and confirmed that, despite numerous attempts, no attacker had succeeded in breaking into it, would your confidence in the vendor's claim be increased, decreased, or unchanged?