Problem 1 Suppose that we removed the %n format specifier from printf implementations, so that %n no longer did anything special. (And suppose we found and fixed any programs that legitimately relied on the %n functionality.)

In this case, would format string vulnerabilities still matter in today’s software environment? Explain.

Problem 2 We trust that the processors in our computers run code faithfully, but it is essentially impossible to verify that a nontrivial piece of silicon will behave according to its specification. We know that processors have shipped with bugs before, because such bugs have been found by users — most famously, the Pentium floating-point division bug of 1994. But what if a processor ships with a malicious backdoor?

a. What is the threat model for malicious processors? Assume that a processor modified by the attacker has made its way into vulnerable computers. How will the attacker access those computers to be able to influence their behavior? How might she want to cause those computers to misbehave?

b. From the attacker’s perspective, the best processor backdoor would allow the attacker (who knows of its existence) to subvert the running system, while not being easily detected in testing.

What is the very simplest backdoor you can think of to a CPU that satisfies these criteria?

Problem 3. Why does the FAA mandate that all airplane bathroom doors have ashtrays? Note that this mandate applies even to new airplanes put into use after all smoking was banned on all U.S. airplanes.

Hint: A fire on board Varig Flight 820, in 1973, which killed 123 passengers, may have started as an improperly disposed-of cigarette.