Song
Spring 2015

CS 161
Computer Security

Discussion 5

Question 1 ***Fuzzing and Symbolic Execution*** **(20 min)**

In this problem, we will explore various approaches to systematically test a piece of code.

```

int foo(uint8_t bar, uint8_t baz)
{
    int buf[500] = {0};
    if ((bar + baz) % 3 == 2)
    {
        buf[(bar + baz) % 500] = 4;
    }
    if (bar > 250 && baz > 250)
    {
        return -1;
    }
    else if (bar > 10 && baz < 245)
    {
        if ((bar % 2 == 0) && (baz % 2 == 1))
        {
            return buf[bar + baz] + 3;
        }
        else if ((bar % 2 == 1) && (baz % 2 == 1))
        {
            return buf[bar + baz + 3];
        }
        else
        {
            return buf[bar + baz];
        }
    }
    else
    {
        return bar + baz + 3;
}
```

Reminder: `uint8_t` is a 1-byte int.

(a) What is the minimum number of test cases required for line coverage?
(b) What is the minimum number of test cases required for branch coverage?
(c) What is the minimum number of test cases required for path coverage?
(d) If we used blackbox fuzzing, what is the probability that a randomly generated set of inputs for bar and baz will cause a buffer overflow?
(e) Write the formula for the values of bar and baz that would cause a buffer overflow.