CS 161 Fall 2006 — Computer Security Joseph/Tygar — HW 3 Solutions

Solutions to exercises:

1. (25 pts.) Digital Millennium Copyright Act

(a) this is clear case against the anti-circumvention part of the DMCA – it involves reverse engineering and changing software and distributing that method.

(b) this does not involve modifying software or digital rights management protection techniques, so it is not illegal

(c) this is controversial, and arguments are on both sides exist, but the primarily favor the legality of such a system

(d) this is clearly legal, existed before the DMCA, and is common practice (also, it has uses other than circumventing DRM)

(e) this is outside the scope of the DMCA – it does not control viewer behavior

2. (25 pts.) Worm Propagation

(a) Using the following is a table of infection fractions, we can see that at time 12, 50 percent of the machines will be infected and at time 24 all machines will be infected.

0 4.1614E-10
1 2.5175E-09
2 1.523E-08
3 9.2136E-08
4 5.5739E-07
5 3.372E-06
6 2.03991E-05
7 0.000123395
8 0.000746029
9 0.004496273
10 0.026596994
11 0.141851065
12 0.5
13 0.858148935
14 0.973403006
15 0.995503727
16 0.999253971
17 0.999876605
18 0.999979601
19 0.999996628
20 0.999999443

21 0.999999908
22 0.999999985
23 0.999999997
24 1
25 1

(b) Using the following is a table of infection fractions, we can see that the 50 percent time has not changed and is still 12. However, the time for all machines to be infected has increased from 24 to 36.

0 2.03991E-05
1 5.01722E-05
2 0.000123395
3 0.000303447
4 0.000746029
5 0.001832939
6 0.004496273
7 0.010986943
8 0.026596994
9 0.062973356
10 0.141851065
11 0.289050497
12 0.5
13 0.710949503
14 0.858148935
15 0.937026644
16 0.973403006
17 0.989013057
18 0.995503727
19 0.998167061
20 0.999253971
21 0.999696553
22 0.999876605
23 0.999949828
24 0.999979601
25 0.999991706
26 0.999996628
27 0.999998629
28 0.999999443
29 0.999999773
30 0.999999908
31 0.999999963
32 0.999999985
33 0.999999994
34 0.999999997
35 0.999999999
36 1
37 1

(c) For the higher infection rate, fewer hosts are infected BEFORE the 50 percent point, than with the lower infection rate. However, after the 50 percent point, with the higher infection rate more hosts are quickly infected.

3. (50 pts.) Buffer overflow exploit

I found where the return address was located on the stack by discovering the address of the instruction in `main()` after the call to `foo()` (with `disassemble`) and then looking for that address after the end of `buf` (with `x`). It turned out to be 76 bytes after `&buf`. I put the shellcode at the beginning of the argument string, and the address I used to overwrite the return address was `arg`. I discovered the address with `print arg` and then put that address at the appropriate place in my string (bytes 77–80).

Solution code for this problem will be available in `/home/ff/cs161/hw3sol/exploit`.