Massachusetts Institute of Technology
6.857: Network and Computer Security
Professor Ronald L. Rivest

Handout 15
November 22, 2006
Due: December 4, 2006

Problem Set 6

This problem set is due *via email,* to 6.857-submit@theory.csail.mit.edu on *Monday, December 4* by the beginning of class.

You are to work on this problem set in groups of three or four people. Problems turned in by individuals, pairs, pentuples, etc. will not be accepted. Be sure that all group members can explain the solutions. See Handout 1 (*Course Information*) for our policy on collaboration. If you do not have a group, let us know.

Homework must be submitted electronically! Each problem answer must appear on a separate page. Mark the top of each page with your group member names, the course number (6.857), the problem set number and question, and the date. We have provided templates for LaTeX and Microsoft Word on the course website (see the *Resources* page).

Grading and Late Policy: Each problem is worth 10 points. Late homework will not be accepted without prior approval.

With the authors' permission, we will distribute our favorite solution to each problem as the "official" solution—this is your chance to become famous! If you do not wish for your homework to be used as an official solution, or if you wish that it only be used anonymously, please note this on your homework.

Problem 6-1. XSS

Cookies are used to keep state information (http is stateless) and sometimes even used to automatically login (whenever you click "remember me"). In this problem you will try to exploit a vulnerability and use it to steal a cookie.

Go to `http://theory.csail.mit.edu/classes/6.857/vulnerable.php` (please do not try this on other pages..). This is a very secure login page for you to access your pset 6 grades.

(a) Give two reasons why this page is actually very insecure, and justify briefly. One of the security holes should help you later. (Do not use automated tools against the page.)

(b) Now suppose that you are a user of the page located at `http://theory.lcs.mit.edu/classes/6.857/cookie.html`. This page stores some sensitive information about you in a cookie. For simplicity this page actually allows you to set the "data" field of your cookie to some value (just click on the "Set Cookie" button). The "Get Cookie!" link is there for you to check. You might need to enable cookies in your browser.

Set "data" to some value, and check that it has been correctly stored.

(c) Leverage the security hole(s) in `vulnerable.php` to steal the cookie, passing it to some other site (say your web.mit page). Explain your method of attack. Why doesn't the same origin policy apply to prevent the steal? Are there some fields in the cookies that you can set to stop this attack?

(d) If a similar vulnerability was found on a web forum, how could you leverage that to steal cookies? Will you need to trick the user into say clicking a link that you crafted?

Problem 6-2. Computer Viruses and Emulation

An important tool in the anti-virus arsenal is the ability to execute programs in a safe hardware emulation mode. Such emulation, for example, is necessary for the *generic decryption* detection schemes currently deployed at the frontlines of the anti-virus battle. In this problem we ask you to explore several issues surrounding emulation and its use to detect malware.

(a) As discussed in class, if a virus can detect that it is being executed in emulation mode, it can defeat generic decryption-style detection by immediately branching back to the normal program. An anti-virus company, therefore, might appreciate *perfect emulation* in which it is impossible for a program to detect whether or not it is running on the actual hardware. List 10 unique ways that a program might be able to detect whether or not it is being emulated. **None of your answers may involve device driver input that is hard to emulate (e.g., looking for keystrokes on the keyboard, specific mouse movements, specific Internet communication, etc.)**

(b) Though perfect emulation might aid virus detection, it also opens up the possibility of helping to support more powerful viruses. For example, earlier this year, Joanna Rutkowska of Singapore-based IT security firm COSEINC claimed to have invented a virus called the "Blue Pill," which transfers the whole operating system (malware-detectors and all) into a virtual machine, allowing the virus to do what it pleases, undetected, in hypervisor mode. [1]

Choose a side in this debate (i.e., perfect emulation would be a useful technology, or perfect emulation is dangerous), and argue your point vigorously. Discuss the pros and cons of both answers, and argue why yours seems the better bet.

(c) Discuss a potential solution (based, perhaps, on special hardware) that would allow users to gain the benefits of both answers from the previous problem part, while minimizing the cons.

Problem 6-3. Aunty Virus

You have just been hired by Aunty Virus Software to help them detect metamorphic viruses: viruses that change their own (decrypted) object code to something different, but equivalent in function.

Your boss, Aunty Vee, directs you to write a program E to tell when two other programs are "equivalent". The plan is to use E to check if a (decrypted) virus body P_1 is equivalent in function to some known virus body P_2, in spite of the virus-writer's efforts to have P_1 look very different from P_2.

Aunty considers programs P_1 and P_2 to be "equivalent" if they behave the same on all inputs. [For the purposes of this problem, assume that all programs take a single input value.] That is, P_1 and P_2 are equivalent if for all inputs x, P_1 on input x terminates if and only if P_2 on input x terminates, and moreover, if they terminate for an input, then their outputs are the same.

(a) Give a careful argument why writing such an "equivalence detector" program E is impossible, using what you have already learned in class. That is, argue that testing the equivalence of programs is undecidable.

(b) Argue that it doesn't matter that equivalence testing is undecidable—even if you had such a program E, virus writers would still be able to escape detection.

(c) How would you argue to Aunty Vee that you shouldn't be fired, since you have now shown what you have been asked to do is both impossible and useless?

[1] c.f., `http://news.yahoo.com/s/zd/20061026/tc_zd/192403`