Massachusetts Institute of Technology — Handout 15
6.857: Network and Computer Security — October 7, 1999
Professor Ronald L. Rivest

# Problem Set 4

This problem set is due on *Thursday, October 14, 1999* at the end of class. Late homeworks will *not* be accepted. You are to work on this problem set in groups of three to four people. (Be sure that you all understand the solutions turned in to each problem.)

Mark the top of each sheet with your names, 6.857, the problem set number and question, and the date. **Type up your solutions**[1], and be clear. Points may be deducted if your TA has problems understanding your solution.

## Problem 4-1. Do You Trust Your Allies?

You are in charge of information security for the DEA. Your agency has been working with British Intelligence to close in on a chain of drug smuggling operations centered in Iceland. The government of Iceland has been very cooperative, and yet lately your sting operations have begun to fail with alarming frequency. Realizing that Iceland has a major economic incentive to warn the criminals before your stings, you conclude that you need to limit the amount and type of information that the Icelandic police agency receives regarding your plans.

Design a command and control ($C^2$) structure to disseminate information to your field agents in a timely fashion, while minimizing risks from OPSEC (operational security[2]) violations. Your structure should also be designed to release sufficient information to partner agencies (such as the Icelandic government) to keep them content, while not endangering your activities or agents, or revealing your mistrust of other agencies.

Issues you might want to consider include (but are not limited to):

- Classification of Data, and defining Need to Know
- Splitting mission information to reduce damage from compromise
- Revealing enemy agents within your organization by data planting

## Problem 4-2. Fun with the Stack

print_display.c:

```
#include <stdlib.h>
#include <stdio.h>

int main() {
  char display[512];
```

[1] If you latex your solutions, include a URL or AFS path where I can find the latex file(s); I will distribute my favorite solution for each problem to the class as the 'official' solution – this is your chance to become famous! Note that if I choose your solution, then you get an extra point on that problem.

[2] OPSEC: The practice of keeping information about day-to-day activities concealed, so that deviations from the norm are not noticed, and to prevent an enemy from attacking where you are repetitive.

```
  strcpy(display, getenv("DISPLAY"));
  printf("Your display environment variable is %s\n",display);

  return(0);

}
```

The system administrator accidentally set the suid (set userid) bit on this program, which is owned by root and lives in /usr/bin/

(a) Describe the security vulnerability present in this program.

(b) Write a program to exploit this vulnerability to obtain a root shell. Be clear and concise. If you use a language other than C or perl, be extra clear.

(c) Argue/demonstrate that your exploit works. Rewrite print_display.c to be safe.

URLS of interest:

- "Smashing the Stack for Fun and Profit":
  `http://www.phrack.com/search.phtml?view&article=p49-14`
- Rootshell exploits: `http://www.rootshell.com/beta/exploits.html`
- The GNU C Library – Users and Groups:
  `http://www.gnu.org/manual/glibc-2.0.6/html_chapter/libc_26.html`