Problem Set 6

This problem set is due on *Thursday, October 26* at the end of class.

Each solution is to be handed in *separately*, i.e. on separate sheets of three-hole punched paper. If the solution to a single problem takes more than one page, staple together only the pages relative to the same problem. Mark the top of each sheet with your name, 6.915, the problem number, and the date.

If you collaborate with other students, you MUST write up solutions on your own and acknowledge the people you work with.

Problem 6-1. User–mountable file systems

In UNIX the `mount` command allows a special file like a floppy disk drive or an alternate disk drive, to be used as a *file system*. Usually the person mounting the file is given root access to the mounted file. This command can be abused to circumvent security systems.

Show what kind of problems can happen if the `mount` command is issued so that the floppy attaches to and thus overwrites an existing directory.

Problem 6-2. Write permission to terminal

The largest percentage of input devices on a UNIX systems are terminals. To aid the programmers and users of these terminals, manufacturers have allowed the keyboards of these terminals to be programmed to new values by sending commands from the computer. To program the function keys in your terminal, you need to have write permission on your terminal. Assume all users have write permission to terminal devices. Show how an user can exploit this in order to gain superuser privileges.

Can you think of a way of exploiting this weakness even without write permission? (*Hint:* A `talk` channel maybe?)

Problem 6-3. Non–atomic updates

In UNIX the `mkdir` command has a special problem called **race condition**. The command creates a directory as a Superuser. This means that the object is owned by the Superuser, instead of the user for which the directory is being created. The `mkdir` command must now change ownership of the created object to that of the end user. There is a small but definite window of opportunity that occurs **between** the creation and the modification of ownership.

Show how this window of opportunity can be exploited.

Problem 6-4. The comand `chroot`

Read the man page of the UNIX command `chroot`. Why is this command restricted to the superuser?

Problem 6-5. Viruses

When a virus infects a computer systems it usually has two goals:

- Infect any other non–infected executable file in the system
- Disguise itself so not to be recognized

Notice that this involves a conflict for the virus. On one had it wants to recongnize itself so not to infect already infected files. On the other hand it does not want to be recognizable by other programs (like a virus scanner.) Comments on the possible ways to "solve" this conflict.