Reminders
- Sign up for mailing list
- Read the web site
- Work through lab information
- Start reading the book
- Do the first homework – due this Monday!
- Read & start project 1

451 Projects
- 3 or 4 projects
- First one – individual, others – groups of 3
- Need basic C and UNIX skills
  - Check links if you need help with this
- Challenging
  - Don’t leave until last minute
- Learn a lot of cool stuff

First Project
- Introduces C and Unix skills you’ll need
- Teaches how to build and run Linux in VMWare
- Two main parts:
  - Write a simple shell
  - Add a simple system call to Linux kernel
- Due: Thursday, Oct 7
  - 1:00 pm for turnin
  - Bring writeup to section

The shell
- Print out prompt
- Accept input
- Parse input
- Create new process
  - Launch specified program there
  - Wait for it to finish
  - Repeat

System Calls
- What’s a system call?
- Examples?
- In your shell:
  - Use fork to create a child process
  - Use execve to execute a specified program
  - Use wait to wait until child process terminates
Adding a System Call

- Add `execcounts` system call:
  - Count number of times you call `fork`, `vfork`, `clone`, and `exec` system calls.
- Steps:
  - Modify kernel to keep track of this information
  - Add `execcounts` to return this to the user
  - Write an application to use it (your shell)

Example of `execcounts`

- `execcounts clear`
- `execcounts dir`
- `execcounts list`
- `execcounts list -for` [file]
- `execcounts --help`

Usage: `exec [program] [options] [arguments]`

Statistics:
- Fork: 3 27%
- Clone: 0 0%
- Vfork: 0 0%
- Exec: 8 72%

Computing Resources

- Develop your code on dedicated 451 Linux hosts:
  - `spinlock`, `coredump`
- Test your code on VMWare PCs in 006
  - Do not use attu

Programming in kernel mode

- Can’t use application libraries (e.g. libc)
  - E.g. can’t use printf
- Use only functions defined by the kernel
  - E.g. use printk instead
- Include files are different in the kernel
- Don’t forget you’re in kernel space
  - E.g. unsafe to access a pointer from user space directly, use fn’s that perform checks
- Good way to learn — look at existing code

VMware

- Software simulation of x86 architecture
- Run an OS in a sandbox
  - Easily reset to known good state

Using VMWare

- All disks are nonpersistent
  - `Powering off loses your changes!` Use “shutdown -r now” instead
- Network adapter is host-only
VMWare basics

- There is only one user: `root`
- The password is `rootpassword`
- You will need to:
  - Build a kernel image on `spinlock/coredump`
  - Transfer it to Linux running inside VMWare
  - Boot Linux in VMWare using your new kernel
- Use ftp to get your files into VMWare
  - FTP to 192.168.93.2 from the host running VMWare.
    - E.g. using IE, go to ftp://root:rootpassword@192.168.93.2

UNIX & C help

- Unix & C tutorial links on 451 projects page
- What if my shell crashes?
  - Use gdb to debug
    - gdb tutorials linked on web site
  - What do I use to compile my shell?
    - gcc
- What do I use to type up my code?
  - I recommend Emacs
- How do I find stuff in the kernel source?
  - Use `grep -r search_string *`
  - Use LXR (Linux Cross Reference): `http://lxr.linux.no/`