Laboratory Assignment #2

Tool Up: Get Familiar with a Kernel/Development Tool

Value: *(See the **Grading** section of the Syllabus.)*

Due Date and Time: *(See the **Course Calendar**.)*

Summary:
The purpose of this assignment is to introduce you to some handy tools to use when developing and debugging Linux kernel code. Linux development is challenging for a number of reasons: lack of built-in debugging environment, hard-to-reproduce bugs, oops messages that can scroll off your screen, and incomplete logging. This assignment will introduce you to a kernel/developmental tool of your choice through the setup and demonstration of your chosen tool to the class. It will also give you a chance to learn about other tools from the presentations of your classmates.

Objective:

- Become more familiar with tools available for Linux development and debugging.

Tasks:

1. Pick a tool from the tool list below, or else email the instructor for clearance to investigate something off the list.
2. Set up the tool on either your machine in MCH 202 or your laptop.
3. Put together a 5-10 minute presentation covering your chosen tool. Be sure to address the following:
    a. Tool's primary purpose/usefulness (as it pertains to Linux kernel programming)
    b. Instructions for setup
        i. If complex, summarize setup and make how-to references available
    c. Tutorial on basic usage
    d. Demo
        i. If a demo is not possible, explain why
        ii. Try to find screenshots online to demonstrate the tool in action
    e. List all references you used

List of Tools:
If the tool may be used with both user-space and kernel-space code, concentrate on setup for kernel programming

- git
- kgdb
- eclipse
- doxygen
- LTTng
- cscope
- etags/ctags
- ftrace

- ktap
- uprobes
- logging kernel messages (e.g., netconsole)
- Code analysis
    - coccinelle
    - Linux Driver Verification
    - …

Delivery Method:

1. Class presentation
2. Presentation materials (e.g. slides, webpage, list of references)
3. Demonstration of tool (if not able to sufficiently demo in-class).