# 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 in 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.
- Decide which tools you will use for assistance in the rest of the programming assignments.

**Tasks:**

1. Pick a tool from the tool list below, or else email the instructor for clearance to investigate something off the list.
2. Record your name and the name of your chosen kernel developmental/debugging tool at the link posted on the Calendar page of the class website . Pick something that has not already been picked. Also choose your presentation timeslot.
3. Set up the tool on either your machine in LOV 016 or your laptop.
4. Put together a 10-15 minute presentation over your chosen tool. Be sure to address the following:
    a. Tool's primary purpose/usefulness
    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:**
See Assignment 2 presentation for additional links and information. If the tool may be used with both user-space and kernel-space code, concentrate on setup for kernel code.

- git
- gdb
- kgdb
- eclipse
- doxygen
- LTTng

- cscope
- etags/ctags

You may also choose to present multiple small tools under a topic:

- Useful ways to catch kernel oops messages
- Useful /proc files (more than what is covered in Assignment 1)

**Delivery Method:**

1. Be sure to sign up for a presentation time during the class presentation week at the Doodle link.
2. Give your presentation.
3. Please send me your presentation materials (e.g. slide show or link to a webpage you set up, list of references)

**Assessment:**

You will receive a perfect sore if you address all points in task #4 in your presentation.

**References:**

- Assignment 2 ppt: http://ww2.cs.fsu.edu/~diesburg/courses/dd/assign/Assignment2.ppt
- Basic debugging ppt: http://ww2.cs.fsu.edu/~diesburg/courses/dd/notes/lecture_4_debugging_techniques.ppt