

**CSCI 5573: Advanced Operating Systems**  
**Fall 2019**

## **Homework Three**

*Due Date and Time: 11:55 PM, Friday, November 15, 2019*

The goal of this homework is to get familiar with Linux Kernel Modules as well as some kernel debugging tools.

You must work in a virtual machine for questions 1 and 2 of this assignment, otherwise you may corrupt your operating system. You will need super user privileges. For questions 3 – 8, your answers should be concise, about half a page per question.

You may work in teams of size two students.

1. In Homework Two, you built a kernel module that can identify all system calls that are being intercepted as well as all new system calls introduced in the kernel after your new kernel module was installed in the kernel. This kernel module runs only when you explicitly invoke it. Now, modify this kernel module so that it periodically checks for any changes in the system call table (no explicit invocation required) and prints out a message on the same tty from which the module was loaded. The output message is either “No changes have occurred in the system call table” or “System call table has been changed”.
2. Write a kernel probe for your kernel module in problem one that dumps the current state of your module before and after a specific instruction of the module is executed. You can choose any instruction of your module to trap. The dumped state should include information such as current values of some registers, process ids, etc.
3. Consider a 16-bit machine running Multics operating system. Describe the steps executed to determine the physical address corresponding a virtual address 2712 of a running process. Assume that the page size is 1 KB and a segment may have up to 16 pages. Point out the steps where a page fault may occur.
4. Explain how capabilities in Hydra provide support for security and protection.
5. Explain how VAX/VMS minimizes the impact of a heavily paging process on other processes running concurrently.
6. Describe two design features of FFS that improve performance by reducing seek times.
7. Describe the structure of a task in Mach operating system in terms of its memory, execution and communication. How are tasks and threads related?
8. Describe two key differences in caching between the Andrew file system and Sprite file system. Describe their consequences.