EE 107S: Introduction to Linux

Course Description

Linux, and commands associated with Linux, are prevalent throughout the ECE industry, from software development to analog IC design. Understanding how to use Linux effectively can improve productivity and reduce the headache of repetition dramatically. In this course we will introduce you to the ideas behind Linux as an operating system and show you how to use some of the core commands available in most Linux systems. The course is not intended to be an in depth study on how Linux works. Instead, we will focus on how to use Linux from a practical standpoint and equip you with the skills to explore deeper.

Prerequisites: None.

Target Audience: Students who have no experience with Linux. No programming knowledge is required, but it may be helpful. Undergraduate status is required.

Instructor: Chirag Sakhuja

Contact: chirag.sakhuja@utexas.edu

Course Objectives

At the completion of this course, students will be able to:

1. comfortably use a Linux command line
2. understand what commands are available and how to learn more about them
3. combine Linux commands to automate complicated tasks

Course Meeting Times

Lecture: T 6:00 - 7:30 pm, EER 1.518

Lab Sections: W 6:00 - 7:30 pm, EER 1.512; Th 6:00 - 7:30 pm, EER 1.518

Course Logistics

Each week, there will be one 1.5-hour long lecture and one 1.5-hour long lab session, for seven weeks total. There will also be weekly assignments (for a total of seven assignments). The course is pass/fail, which will be determined by the completion of the assignments. To pass, you must receive an overall assignment average of at least 50%, and you may miss at most two assignments. Attendance to lecture and lab is encouraged, but not mandatory, other than the first lab session, which counts as the first assignment.

Advising Notes

This course can only be taken on a pass/fail basis and cannot count towards the academic enrichment technical core. More specifically, the EE 107S course does not count towards the ECE major, and has 0 hours of engineering topics and 0 hours of math and basic science for the purposes of ABET metrics.

Lectures

Each lecture we will consider a single topic, such as remote access or text processing. We will look at several commands related to the topic and then use the commands to solve a small problem. Some examples are:

- Managing a remote session so that you can disconnect and reconnect to a remote machine without losing your work
- Downloading an open-source project and building it into an executable
- Formatting and extracting information from a course roster

The lectures will be interactive, so please bring your laptops to class!

Tentative Course Outline

Week	Content
Week 1	• Course outline and overview • What is Linux and how is it used? • Set up a Linux environment • Introduction to shells (we will be using Bash) • File system navigation and structure (`ls`, `pwd`, `cd`, `mkdir`, `rm`, `cp`, `mv`) • **Assignment:** Look through man pages for each of the commands covered; attend the first lab section!
Week 2	• Creating simple files (`touch`, `echo`) • Viewing files (`cat`, `head/tail`, `less`) • Pipes and I/O redirection • Searching and processing files (`find`, `grep`, `tr`, `cut`, `awk`) • **Assignment:** Extract links from an HTML file and output a CSV
Week 3	• Introduction to Bash scripting • Editing text from the command line (`vim`) • **Assignment:** Compare `vim` and `emacs` and familiarize yourself with one (or both)
Week 4	• Accessing remote machines (`ssh`, `scp`) • Managing multiple shells in a single session (`tmux`) • Forwarding a display from a remote machine (`vncserver`) • Customize tools with dotfiles • **Assignment:** Create a personalized .vimrc/.emacs and .tmux.conf; set up a reverse tunnel to enable VNC from outside a network or SCP from the remote-side command line
Week 5	• File permissions and security (`chmod`, `sudo`) • Package managers and installing new software (`apt`) • Building and installing from source (`make`, configure scripts) • **Assignment:** Build and install an open source project without superuser permissions
Week 6	• Introduction to version control • Basics of `git` • **Assignment:** https://github.com/git-game/git-game
Week 7	• File compression (`zip`, `tar`) • Bash scripting, part 2 • Managing multiple jobs (`jobs`, `bg`, `fg`) • Managing processes (`ps`, `htop`, `kill`) • Miscellaneous topics (networking, debugging, etc.) as time permits • **Assignment:** Create a script that runs multiple instances of a program in parallel with different command line arguments