Linux Shell Programming

Melvyn Drag

Spring 2019

E-mail: `mdrag1@njcu.edu`
Office Hours: Th 6-7 & Fri 10-1
Office: CS Adjunct Office

Web: **melvyniandrag.github.io**
Class Hours: Th 7-9:45pm
Class Room: TBD

Course Description

In this fun, fast-paced class you're going to become a "Linux Rockstar". Within a few short months you'll get lots of great information to make you more employable and more efficient on Linux machines. We will learn about *nix operating systems and programming in this environment. When possible we will explore topics in depth. More complex topics will be briefly touched upon to acquaint you with important *nix vocabulary words and key concepts.

Required Materials

- Access to a Google Cloud Virtual Machine (provided and free).

Course Objectives

You will master many aspects of Linux/Unix, and acquaint yourself with others. You will learn:

1. bash
2. vim
3. awk
4. sed
5. network programming
6. file system management
7. version control
8. webservers
9. Linux account management
10. run scheduled tasks
11. (a little) C.
12. kernel driver development
13. containerization/virtualization.
14. ...and more...

Course Structure

Class Structure

This material will likely be new to you, and it's very interesting. There will be weekly lectures and assignments as well as two exams. Every week we will cover a new aspect of Linux / Unix / Programming / System Administration and there will be a related assignment due the following lecture. In the event that you miss a class there will be related reading materials posted online so you can still complete your assignment.

Lecture

Each week in class we will log on to our virtual machines and explore a specific aspect of programming in Linux.

Assessments

Weekly programming assignments based on the materials covered in class. 2 exams bringing together various aspects of topics taught. No late assignments. Lowest weekly assignment grade will be dropped. Multiple choice midterm and final exams.

Grading Policy

Grading is as follows:

- **20%** Midterm exam.
- **20%** Final exam.
- **60%** Average of weekly assignments.

For example, if you get an 80 on exam 1, an 85 on exam 2, and (to simplify assume there were only two weekly assignments) your weekly assignment scores are [90, 95], your final score will be 20pts * 0.80 + 20pts * 0.85 + 60pts * (0.90 + 0.95) / 2 = 88.5 pts, and will be translated to the appropriate letter grade following NJCU guidelines.

Schedule and weekly learning goals

The schedule is tentative and subject to change. The learning goals below should be viewed as the key concepts you should grasp after each week, and also as a study guide before each exam.

Date: 01/24 - Linux + BASH + Vim Part 1

- Connect to cloud machine
- What is Linux, different distros.
- Learn basic BASH commands:

    1. ls (-l -la -a)
    2. hidden files
    3. cd
    4. cp
    5. mv
    6. whoami
    7. which
    8. mkdir
    9. touch
    10. rm
    11. touch
    12. cat (assignment1)
    13. echo (assignment1)
    14. man
    15. wc (assignment1)
    16. md5sum (assignment1)
    17. df
    18. du
    19. uniq
    20. sort
    21. find
    22. xargs
    23. more
    24. less
    25. grep (no regex yet)
    26. unzip (assignment1)
    27. tar
    28. cut (assignment1)
    29. if (assignment1)
    30. arrays
    31. Return values and $? (assignment1)
    32. tree
    33. etc.

- wildcards
- bash variables (export vs without export and subshells)
- bash eq, not equal
- vim (modes, p, shif+p, :wq, :x, i, a, dd, arrows, <esc>, skipping around. set nu/nonu. ctrl+v. search+replace s///g s///gc. :sort. :sort n. u for undo, ctrl+r. ctrl+a to increment. split vs vsplit. +/- for next line. 5w = 5 words. 5l = 5 letters.)
- pipes | (assignment1)
- telnet starwars
- csh, zsh, sh, dash, etc.

Assignment: Program to download a zip, check an md5sum, and output the number of words in the text files in the zip. Exit script early if there is an error.

Date: 01/31 Bash + Vim Part 2

- grep
- regular expressions
- nohhup
- history | grep
- rerouting output with > < etc.
- discuss fd.
- /proc/$PID/fd
- jobs
- stderr, stdout, stdin
- top, htop
- tee
- bg, fg
- ps
- kill
- bc. Math in bash.
- Bash functions

Assignment:

1. Write a few regular expressions to find pids of jobs currently running.

Date: 02/7 - sed and awk

- awk
- sed

Assignment:

1. Write an AWK program to get the largest value in a multicolumn file with a standard delimiter
2. Write a sed program to replace instances of a pattern with a new field.

Date: 02/14 Users, Manage Privileges, Kernel, Other Unix commands

- add/delete users
- create/delete groups
- password management
- root, sudo, etc.
- cron
- chmod, chgrp
- user limits

Assignment:

1. Implement a way to stop all of user $USERNAME's processes at 9AM and restart them at 5PM

Date: 02/21 - Networking

- ssh
- iptables
- ufw
- netstat
- netcat
- (extra miscellaneous topic) Hard links vs soft links

Assignment:

1. Write a few netstat commands
2. Build a chat application with netcat to talk to a friend.
3. allow ssh into machine on a port other than 22

Date: 02/28 Running a web application

Make a hello world Flask application, access with browser and cURL.

- Run webserver
- Access web site with browser
- Use of cURL
- Apache vs nginx vs others

Assignment:

1. Get a Flask REST API on your machine I can hit for information about you.

Date: 03/07 - Filesystems

- Linux file system hierarchy. Look at /proc, /bin, /opt, etc.
- GPT vs MBR
- BIOS vs UEFI
- different file system types
- mount/umount
- partitions vs disks
- partition with parted
- partition with fdisk

Assignment:

1. Move your home directory to a new partition. Mount on boot with /etc/fstab.

Date: 03/21 Maintenance and security

- backups
- PAM
- UNIX password management, shadow, password
- AppArmor
- SELinux

Assignment:

1. Set up regular backups on your machine.

Date: 03/28 git

- version control
- git
- git servers
- add, commit, branch, merge, push , pull, remotes, clone

Assignment:

1. Configure your machine as a git server so I can clone / pull / push to verify functionality.

Date: 04/04 C Programs

- gcc, clang
- system calls
- C
- C++
- signal handling (revisit kill command)
- setuid/setgid/stickybit

Assignment:

1. Write, compile and set permissions on a simple C program that can run root commands as a nonprivileged user.

Date: 04/11 Package Mgmt, Virtualization, Containerization

- apt, yum
- brief mention of pip, virtualenvs in Python
- Docker vs. VMs
- Install docker, run a container

Assignment:

1. Run a simple Flask web application in a Docker container on your machine.

Date: 04/18 Operation of running systems

- SysV vs. Systemd and the drama surrounding.
- (if time) upstart
- Bootloaders
- Startup/shutdown
- Run levels

Assignment:

1. Configure a job to run at startup using systemd
2. Configure a job to run at startup using sysvinit
3. (if time) Configure a job to run at startup using upstart.

Date: 04/25 RAID and LVM

- Software RAID
- Hardware RAID
- LVM

Assignment:

1. Configure RAID1/RAID5 on google cloud.
2. Setup LVM across multiple disks.

Date: 05/02 Network Programming

- Beejs networking tutorial
- ports, sockets, ipaddresses, etc.

Assignment:

1. Fix a broken network program by changing blocking/nonblocking sockets.

Date: 05/09 Kernel + Driver Development

- Upgrading kernel in centOS7
- insmod, rmmod
- Write a simple driver
- Discussion of userspace vs kernel space

Reading:
https://www.tldp.org/LDP/lkmpg/2.6/html/c119.html

Assignment:

1. Write a small kernel module.