

# Linux Kung Fu

---

Stephen James  
UBNetDef, Spring 2017



# Introduction

---

What is Linux?

What is the difference between a client and a server?

# What is Linux?

---

- Linux generally refers to a group of Unix-like free and open-source operating system distributions that use the Linux kernel.
- Examples of Linux distributions:
  - CentOS
  - Debian
  - Fedora
  - Linux Mint
  - Red Hat Enterprise Linux (RHEL)
  - Ubuntu

# Servers versus Clients

---

- Servers provide services
- Examples:
  - Database server
  - File server
  - Mail server
  - Web server
- Clients consume services
- Examples:
  - Your personal computer
  - Library computers
  - Your cellular phone

# The Terminal

---

# The Terminal

---

- Your shell prompt can be a useful source of information.
- The shell prompt can be customized.
  - This can be done by changing the variable `$PS1`.
- You enter commands in the terminal.

# The Terminal

---

- `you@ubnetdef:~$`
  - Username: you
  - Host name: ubnetdef
  - Current working directory: ~
  - Superuser: No (\$)
- `root@universe:/etc/init.d#`
  - Username: root
  - Host name: universe
  - Current working directory: /etc/init.d
  - Superuser: Yes (#)

# Basic Linux Commands

---



\$ pwd

---

- The pwd command prints the name of the current working directory.
- Essentially, it tells you where you are.
- \$ sjames5@ubnetdef:~\$ pwd
  - /home/sjames5

# \$ echo

---

- The echo command echoes (or displays) text.
  - `$ echo "I love the terminal!"`
- The text is sent to standard output by default, but can be redirected.
  - `$ echo "Why did you redirect me?" > redirect.txt`

```
$ clear
```

---

- The clear command clears the terminal's screen if possible.

# \$ ls

---

- The ls command lists the contents of a directory.
  - \$ ls
  - \$ ls /etc
- To include hidden entries:
  - \$ ls -a
  - \$ ls -A
- Did you want more information?
  - \$ ls -l
- They can even be used together!
  - \$ ls -Al /var

# \$ cd

---

- The cd command can be used to change your current working directory.
  - \$ cd ..
  - \$ cd /var/log
- Special directory paths:
  - . - the current directory
  - .. – the parent directory
  - ~ - the current user's home directory

# \$ history

---

- Yes, your commands are being logged!
- Essentially, the history command allows you to see the last commands used by the current user.
- The file `~/.bash_history` will be updated at the end of your session (when using Bash).
  - You can bypass this by clearing the current session's history:
    - `$ history -c`

# \$ cat

---

- The cat command concatenates files and/or standard input, printing the result to standard output (by default).
  - \$ cat file1.txt file2.txt file3.txt

## \$ more

---

- The more program is a file pager.
- It allows you to read files, with support for scrolling down.



# \$ less

---

- Less is the opposite of more. Seriously, it is.
- The less program has more features than the more program.
- The less command gives a terminal pager that allows you to view files with support for scrolling up and down.
- `$ less filename.txt`

# \$ mkdir

---

- The mkdir command can be used to make directories.
- To make a directory:
  - \$ mkdir <directory-name>
    - \$ mkdir test
- To make a directory (creating parent directories if needed):
  - \$ mkdir -p <directory-name>
    - \$ mkdir -p this/is/a/test

# \$ rm

---

- The rm command can be used to remove files or directories.
- To remove a file:
  - `$ rm random_file.txt`
- To remove any directory:
  - `$ rm -rf random_dir/`
- Note: The rm command will remove without warning, so be careful (especially with -rf).
- To remove an empty directory:
  - `$ rmdir empty_dir`

# \$ man

---

- The man command provides an interface to reference manuals.
  - \$ man pwd
  - \$ man man
- For a shorter response, you can often use the --help flag:
  - \$ touch --help

# User Management

---

# Users and Groups

---

- Create a user account:
  - # adduser <username>
- Create a group:
  - # addgroup <groupname>
- Add a user to a group:
  - # usermod -G <groupname> -a <username>

# Users and Groups

---

- See all groups a user is in:
  - `$ groups`
  - `$ groups <username>`
- See more information about a user:
  - `$ id`
  - `$ id <username>`
- See the following files:
  - `/etc/passwd`
  - `/etc/groups`

# \$ passwd

---

- The passwd command allows changing the passwords of user accounts.
- Changing user passwords:
  - \$ passwd
  - # passwd <username>
- Locking and unlocking user accounts:
  - # passwd -l <username>
  - # passwd -u <username>
- The passwords are stored as hashes in the file /etc/shadow



\$ su

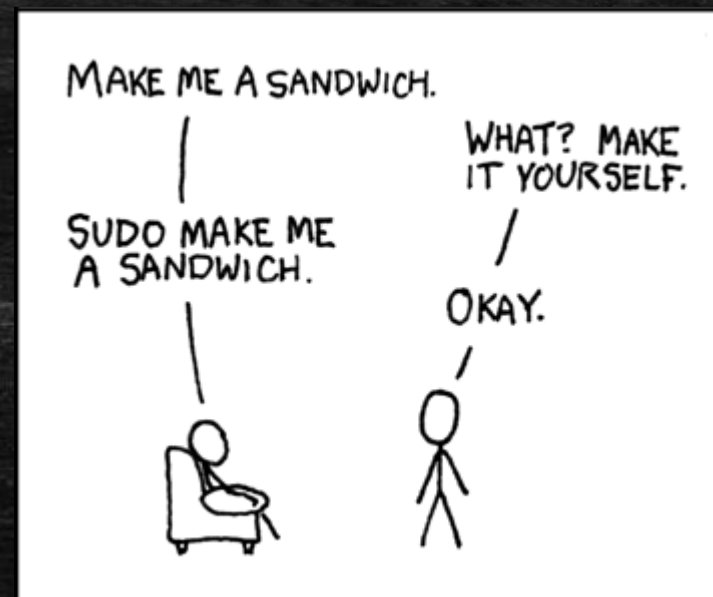
---

- The su command allows you to switch user.
- If no username is specified, the superuser account (root) will be used by default.

# \$ sudo

---

- Allows permitted users to execute a command as the superuser (i.e. "superuser do") or another user (if specified).
- Configured in the file `/etc/sudoers` (can be edited with the `visudo` command)
  - `# visudo`
- Examples:
  - `$ sudo whoami`
  - `$ sudo cat /etc/sudoers`



# Processes and Networking

---

# \$ ps

---

- The ps command provides a process snapshot.
- \$ ps
- \$ ps aux

## \$ top

---

- Similar to ps, but is interactive and updates every second.
- A similar utility, htop, provides a similar function, but usually needs to be installed first.

# \$ kill

---

- To ask a process to ~~commit suicide~~ terminate (but it could choose to ignore you):
  - \$ kill <pid>
- To ask the kernel to ~~commit homicide~~ kill a process (this cannot be ignored):
  - \$ kill -9 <pid>
  - \$ kill -KILL <pid>
  - \$ kill -SIGKILL <pid>

# \$ ping

---

- The ping command sends an ICMP ECHO\_REQUEST to network hosts.
- Pinging IP addresses is usually a simple way to check if your internet connection is working.
- Example:
  - \$ ping 8.8.8.8

# \$ ifconfig

---

- The ifconfig command can be used to view or configure network interfaces.
- View all interfaces:
  - \$ ifconfig
- View specific interface:
  - \$ ifconfig <interface-name>
    - \$ ifconfig lo
- Bring an interface online or offline (respectively):
  - # ifconfig <interface-name> <up | down>
    - # ifconfig ens32 up



Services

---

# Services

---

- In Linux, services are applications or processes that run in the background.
- They are sometimes referred to as daemons.
  - Many of their names will end with “d” out of convention (e.g. sshd, httpd, mongod).

# Services

---

- There are two main ways to control services:
  - System V (older; also called SysV)
  - systemd (newer)
- System V
  - `# service <name> <start | stop | restart | reload | status >`
    - `# service sshd status`
- systemd
  - `# systemctl <start | stop | restart | reload | status > <name>`
    - `# systemctl reload nginx`

# Package Managers

---

# Package Managers

---

- Package managers can help with automating common tasks such as installing, upgrading, and uninstalling programs or *packages*.
- Examples:
  - apt (Advanced Packaging Tool)
    - apt-get
  - aptitude
  - dpkg
  - yum (Yellowdog Updater, Modified)
  - dnf (Dandified Yum)

# \$ apt

---

- Update the local package index:
  - # apt update
- Upgrade a package:
  - # apt upgrade <package-name>
- Upgrade all packages:
  - # apt upgrade

# \$ apt

---

- Install a package:
  - # apt install <package-name>
- Uninstall a package (leave configuration):
  - # apt uninstall <package-name>
- Uninstall a package (remove configuration):
  - # apt purge <package-name>
- Uninstall unneeded dependencies:
  - # apt autoremove

# Useful Tips and Tricks

---



# Useful Tips and Tricks

---

- Pressing the up arrow recalls the previous commands.
- Pressing tab while typing a command can sometimes help to autocomplete a command's name or a file/directory path.
- If you need to stop a currently-running command, use Ctrl+C.
- Typing "!!" in the terminal will re-run the last command.
- If you accidentally print the contents of a binary file to the terminal, it may affect the terminal's display. The "reset" command can be used to resolve that issue.

# Directory Structure

---

# Directory Structure

---

- /bin: Contains all binaries that are necessary for the system to function
- /boot: Contains the linux bootloader
- /dev: Contains all raw DEVICES
- /etc: Contains configuration files for the system
- /home: Contains user's home directories
- /mnt: Typically used when mounting devices (like a cdrom)

# Directory Structure

---

- /opt: Optional, add-on packages
- /proc: A pseudo-file system containing information about processes
- /root: User root's home directory
- /sbin: Contains all the binaries that are necessary for the system to run. Only super users can use these.
- /tmp: Temporary files
- /usr: User-installed files

Want More Commands?

---

# \$ lsof

---

- The lsof command provides a way to list open files.
- All open files:
  - \$ lsof
- All open files by a specific user:
  - \$ lsof -u <username>
- All open network connections:
  - # lsof -i

# \$ route

---

- The route command provides a way to view and manipulate the routing table.
- To view the routing table:
  - \$ route
- To specify 10.0.0.1 as the default gateway:
  - # route add default gw 10.0.0.1

# \$ dig

---

- The dig command is a DNS lookup utility.
- This is useful for verifying that DNS records are being properly resolved.
- This utility disregards the /etc/hosts file.
- Example:
  - \$ dig ubnetdef.org
  - \$ dig +short ubnetdef.org