Unix: A Simple Operating System

John Dempsey
COMP-232: Programming Languages
California State University, Channel Islands
August 25, 2025
Hard Due Date: September 5, 2025

"Unix is basically a simple operating system,
but you need to be a genius to understand the simplicity."
Dennis Ritchie

For this assignment, you just need to open a Unix session and type in the following Unix commands. To turn in this assignment, we will upload the lab1.txt file to comp232.com during LAB 3.

% **pwd** ← Print working directory.

% **mkdir LAB1** ← Make directory for LAB1.

% **cd LAB1** ← Change directory into LAB1.

% **script lab1.txt** ← **Records the commands you type into file lab1.txt.**

Script started, file is lab1.txt ← **Make sure you see that the script started.**

% **ls -l lab1.txt** ← You should see your lab1.txt file.

% **whoami** ← This command tells you who you are. Now you know. 😉

% **date** ← Displays today's date and current time.

% **df** ← Disk free. Shows how full your disk drive is.

% **uptime** ← Displays how long the system has been up, number of users, and system load averages for the past 1, 5, and 15 minutes. Type 'q' to quit.

% **top** ← Shows which processes are using the most CPU on your system.

q ← Type 'q' to quit top

% **id** ← Shows your user id, current group id, and all Unix groups you're in.

% **pwd** ← Print working directory (pwd). Shows you what directory you're in.

% **echo $USER** ← Displays your user login

% **echo $SHELL** ← Displays what Unix shell you are using.

% **grep $USER /etc/passwd** ← Displays your user entry in the /etc/passwd file.

Here's what I see:

```
john@oho:~$ grep $USER /etc/passwd
john:x:1000:1000:,,,:/home/john:/bin/bash
```

john – Your user id.
x – Previous location to store your password.
1000 – Your numeric user id
1000 – Your numeric group id
/home/john – Your home directory
/bin/bash – The shell you’re using when you login.

% **grep mail /etc/passwd** ← Shows mail user. Note that mail user cannot log into system.

% **cat /etc/passwd** ← Displays the entire content of the file, in this case /etc/passwd file.

% **more /etc/passwd** ← Display all user entries in /etc/passwd page by page. Press RETURN to go to next page and q to quit.

% **head -5 /etc/passwd** ← head -5 displays the top 5 lines of a file.

% **tail -10 /etc/passwd** ← tail -10 displays the last 10 lines of a file.

% **sort /etc/passwd** ← Sorts the password file.

% **touch myfile** ← Creates a file of zero length called myfile.

% **ls -l myfile**

Here’s what I see on my system:

```
john@oho:~/test$ ls -l myfile

-rw-r--r-- 1 john john 0 Jan 27 10:57 myfile
```

Let’s explain this:
The first dash can be – or d. Files are ‘-‘ and directories start with a ‘d’.
The next three characters are “rw-“ explains your user access. In this case, you have read and write acess, but cannot execute (or run) this file.
The next three characters “r—” are group access. The “john” group can read this file.
The last three characters “r—” are world access. Everyone on the system can read this file.
0 indicates the size of this file in bytes. Touch does not create any content, so the file is 0 bytes.
Jan 27 10:57 – This file was created on January 27 at 10:57 AM
myfile – This is the name of the file.

% **more /etc/group** ← Displays all groups defined on your system.

% **grep <userid> /etc/group** ← Lists the Unix groups you are in. You’re in the audio group.

% **chgrp audio myfile** ← Change group for myfile to audio.

% **ls -l** ← Note the group is now audio

% **mkdir mydir** ← Make directory (mkdir) called mydir.

% **ls -l mydir**	← Shows that nothing is in the mydir directory.
% **ls -ld mydir**	← Shows the owner, group, and permissions for mydir
% **ls -l ..**	← Lists all files and directories above your current directory.
% **ls -l ../..**	← Lists all files/directories two level above your current directory.
% **cd mydir**	← Change into mydir directory.
% **pwd**	← Print working directory.
% **echo "Hello World!" > hello.txt**	← Writes "Hello World!" to the hello.txt file.
% **cat hello.txt**	← Display hello.txt content.
% **echo "Hello Again!" >> hello.txt**	← Appends "Hello Again!" to the hello.txt file.
% **cat hello.txt**	← Displays content of hello.txt file. Should see two lines.
% **cat h<TAB>**	← Type "cat h" then hit TAB key. TAB should auto fill hello.txt.
% **cp hello.txt mycopy.txt**	← Copies hello.txt to mycopy.txt
% **echo Goodbye >> hello.txt**	← Appends Goodbye to the end of hello.txt
% **more *.txt**	← Displays hello.txt and my
% **diff hello.txt mycopy.txt**	← Displays any differences between two files.
% **ls -l**	← List details and sizes for both files.
% **wc *.txt**	← Displays the number of lines, words, and characters in each file.
% **file *.txt**	← Tells us what kind of file it is, like ASCII text.
% **file /bin/ls**	← Tells us what kind of file it is, like "ELF 64-bit LSB" which is an executable file in Executable and Linkable Format, 64 bit, and Linux Standard Base.
% **cp hello.txt goodbye.txt**	← Copies hello.txt to goodbye.txt.
% **ls -l *.txt**	← Lists details on hello.txt and mycopy.txt
% **rm mycopy.txt**	← Removes file mycopy.txt.
% **ls -l *.txt**	← No more mycopy.txt.
% **pwd**	← Display working directory
% **cd**	← Change to your home directory
% **pwd**	← Display working directory.
% **ls -la**	← List all files and directories in your home directory.

% **ls -laR** ← List all files/directories recursively.

% **cd /etc** ← Change to the /etc directory.

% **ls** ← List all files in the /etc/directory.

% **cd -** ← Change back to previous directory.

% **pwd** ← Display current directory.

% **ls** ← List files in current directory.

% **cd -** ← Change back to /etc directory.

% **pwd** ← You're in /etc directory.

% **cd -** ← **cd –** is a geat command to remember!

% **history** ← Display commands you recently typed.

% **!<number>** ← Rerun any command in list by selecting its number, like: !50.

Select the up/down arrow keys on your keyboard to select a command. Hit ENTER to run.

% **<UP/DOWN ARROW KEYS>** ← Use the UP and DOWN keys on your keyboard to select the previously run command above: **echo Goodbye >> hello.txt**

% **env | more** ← Displays all environment variables, like $PATH.

% **alias** ← Display all aliases. You can define your own aliases.

% **alias lr='ls -lR'** ← Create an alias called lr.

% **lr** ← Run lr.

% **man man** ← Displays manual page for the man command. Press q to quit.

% **man ls** ← Displays all options for the ls command.

% **man fprintf** ← Displays fprintf usage used in C programs.

ho

Unix Commands for Networking

% **hostname** ← Display your hostname

% **ping <hostname>** ← Ping your computer. Sends a short 64 bytes to see if host is up.

To stop the pings from running, type: CONTROL-C (Press CTRL and C buttons at the same time.)

% **ping localhost** ← Ping your computer. Note 127.0.0.1 is being pinged.

% **domainname** ← Display your domainname

```
% ping plus1se.com             ← Ping your instructor's website

% ping todaywebuy.com          ← Ping your instructor's website

% ping whitehouse.gov          ← Ping the White House

% traceroute whitehouse.gov    ← When it works, this is a very cool command to use.
                                 Displays hosts used to reach whitehouse.gov.
                                 You can ^C to stop traceroute.

% nslookup                     ← Let's query the internet using nslookup

> repoleaf.com                 ← Display repoleaf.com's IP address.

> whitehouse.gov               ← Display whitehouse.gov's IP address.

> exit                         ← Exit nslookup.

% dig whitehouse.gov           ← dig whitehouse.gov.

% dig whitehouse.gov MX        ← Display the mail server for whitehouse.gov.

% dig whitehouse.gov TXT       ← Display whitehouse.gov DNS TXT fields.

% dig openhouseon.com TXT      ← Display openhouseon.com's DNS TXT fields.

% netstat                      ← Displays internet connections.  (There are none.)

% ifconfig -a                  ← Displays the following text:
```

```
        john@oho:/etc$ ifconfig -a

        Command 'ifconfig' not found, but can be installed with:

        sudo apt install net-tools
```

```
% sudo apt install net-tools   ← Install net-tools, which will include the ifconfig command.

% ifconfig -a
```

This command displays networking information for each network interface on your system. eth0 is for ethernet connection. lo is for loopback. wifi0 is for your first wifi interface. wifi1 your second wifi interface.

Here's what my system shows for my eth0 interface:

```
john@oho:/etc$ ifconfig -a

eth0: flags=64<RUNNING>  mtu 1500
inet 169.254.36.36  netmask 255.255.0.0
inet6 fe80::ec81:97e7:28ab:2424  prefixlen 64  scopeid 0xfd<compat,link,site,host>
ether bc:17:b8:cf:ef:21  (Ethernet)
RX packets 0  bytes 0 (0.0 B)
RX errors 0  dropped 0  overruns 0  frame 0
TX packets 0  bytes 0 (0.0 B)
TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

When at work, you'll need to know your IP address. From the ifconfig command, we can see that my IP address is 169.254.36.36, my netmask is 255.255.0.0, and my MAC address is bc:17:b8:cf:ef:21.

Combining Unix Commands Using the Pipe Symbol, |.

% **ifconfig -a | grep inet** ← Display all inet addresses

% **ifconfig -a | grep inet | grep -v inet6** ← Display only inet (not inet6) addresses.

% **cat /etc/group | grep <username>** ← Display which groups you are in.

% **cat /etc/group | grep <username> | sort** ← Display which groups you are in sorted order.

% **cat /etc/group | grep <username> | wc -l** ← Displays the count of how many groups you're in.

% **ps -ef** ← Displays all processes running on your system.

% **ps -ef | grep <username>** ← Displays your running processes.

Almost Done!

% **cd** ← Return to your home directory

% **cd LAB1** ← Change into LAB1 directory.

% **ls -ld my*** ← List all files and directories starting with my

Here's what I see:

```
john@oho:~/LAB1$ ls -ld my*
drwxr-xr-x 1 john john  4096 Jan 27 12:18 mydir
-rw-r--r-- 1 john audio    0 Jan 27 12:12 myfile
```

% **date** ← Display current date and time.

% **exit** ← Exits from the script command (the first command we typed.)
Should see “Script done, file is lab1.txt”

% **more lab1.txt** ← Should contain output for all commands you ran.

During LAB 3, I’ll explain how to turn in your LAB 1 and LAB 2 work during the lab.