Tech Education Spring 2024
Homework 1: Exploring Linux
Written by Archer Heffern

HW1:

Goal: Understand what your operating system has to offer - what programs are available, system calls, and how to read documentation!

Why:

- Understand what the OS can do
- All of your development will be done using an operating system, learn how to use it!
- Every TAMID Project I have worked on required some understanding of OS, how to debug, and how it works

1. Read documentation:

Knowing how to efficiently parse information is a critical skill as a developer, aka don't read the docs like it's a book! Get familiar with the common sections, where they usually are, and which you should look at first.

a.

Run `man <section> <name>` in your terminal to learn about the following topics.

- man
- intro(1)
- intro(2)
- hier(7)

eg. `man 2 intro` for intro(2)

and follow the man pages listed in the "SEE ALSO" section for all of these manpages.

b.

Get familiar with the system calls Linux exposes and get a feel for what the OS can actually do!
https://man7.org/linux/man-pages/man2/syscalls.2.html

Tips for reading man pages:

`/<word>` to search (case sensitive)
`man -k ...` to search by regex (string pattern matching)
`man -K ...` to search for text in all manual pages (not just page name)

If you want to see all the manpages, check `man man` and find the FILES section for the physical location of all the manpages.

Tech Education Spring 2024
Homework 1: Exploring Linux
Written by Archer Heffern

Processes

Here is an amazing video explaining this approach to learning Linux. I would definitely watch this before proceeding.

The operating system is running a lot of processes. Let's check them out!

Run any of the following commands:

`ps aux`

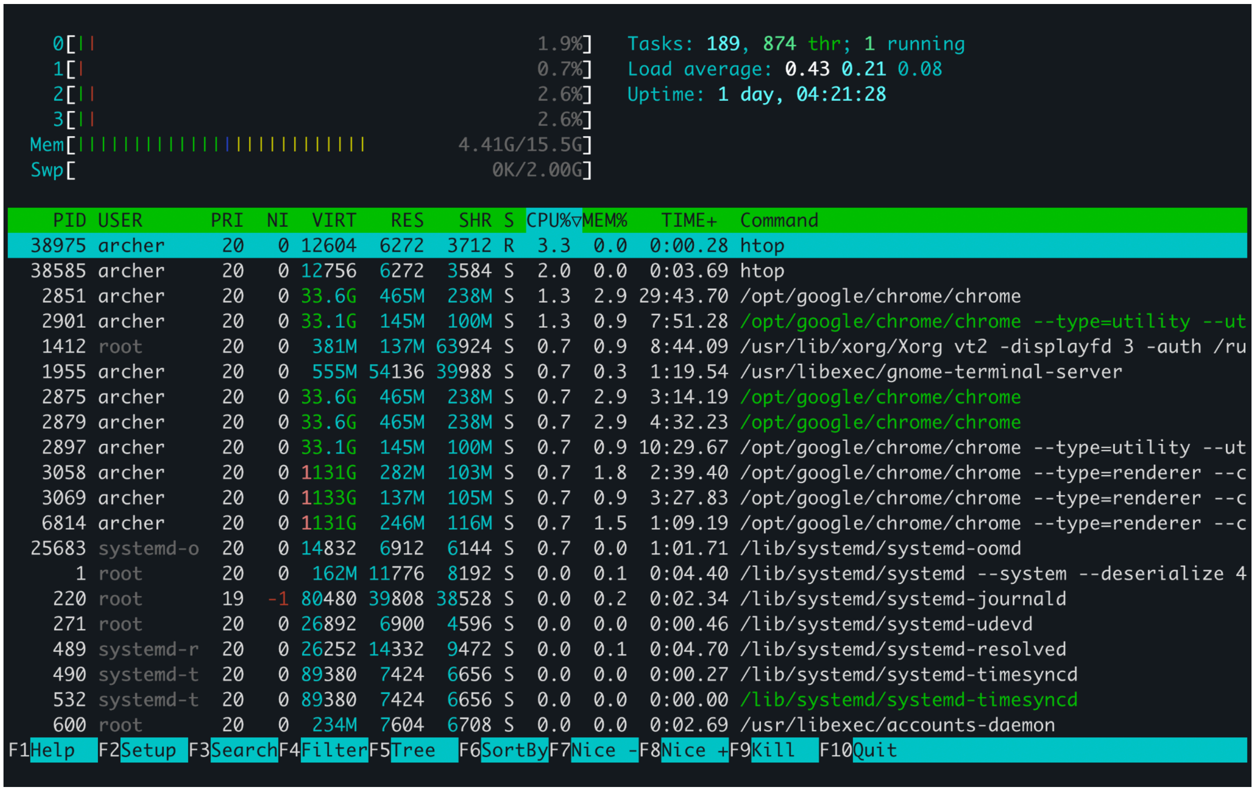
`top` or `htop`

`pstree`

These are all commands, and as such they have manpages! Read them to learn more.

They will display running processes, look at the names of programs and search them up on google.

Example:



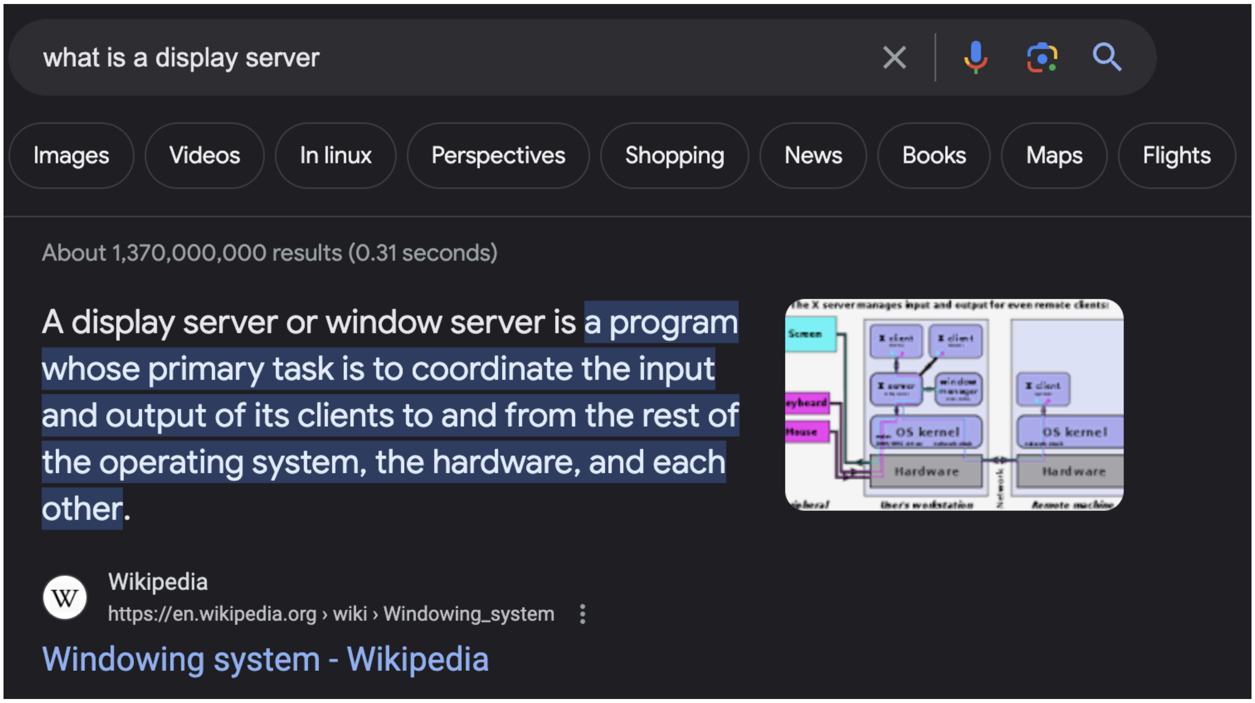
```
    0[||                                      1.9%]   Tasks: 189, 874 thr; 1 running
    1[|                                       0.7%]   Load average: 0.43 0.21 0.08
    2[||                                      2.6%]   Uptime: 1 day, 04:21:28
    3[||                                      2.6%]
  Mem[||||||||||||||||||||||||||       4.41G/15.5G]
  Swp[                                    0K/2.00G]

    PID USER      PRI  NI  VIRT   RES   SHR S CPU%▽MEM%   TIME+  Command
  38975 archer     20   0 12604  6272  3712 R  3.3  0.0  0:00.28 htop
  38585 archer     20   0 12756  6272  3584 S  2.0  0.0  0:03.69 htop
   2851 archer     20   0 33.6G  465M  238M S  1.3  2.9 29:43.70 /opt/google/chrome/chrome
   2901 archer     20   0 33.1G  145M  100M S  1.3  0.9  7:51.28 /opt/google/chrome/chrome --type=utility --ut
   1412 root       20   0  381M  137M 63924 S  0.7  0.9  8:44.09 /usr/lib/xorg/Xorg vt2 -displayfd 3 -auth /ru
   1955 archer     20   0  555M 54136 39988 S  0.7  0.3  1:19.54 /usr/libexec/gnome-terminal-server
   2875 archer     20   0 33.6G  465M  238M S  0.7  2.9  3:14.19 /opt/google/chrome/chrome
   2879 archer     20   0 33.6G  465M  238M S  0.7  2.9  4:32.23 /opt/google/chrome/chrome
   2897 archer     20   0 33.1G  145M  100M S  0.7  0.9 10:29.67 /opt/google/chrome/chrome --type=utility --ut
   3058 archer     20   0 1131G  282M  103M S  0.7  1.8  2:39.40 /opt/google/chrome/chrome --type=renderer --c
   3069 archer     20   0 1133G  137M  105M S  0.7  0.9  3:27.83 /opt/google/chrome/chrome --type=renderer --c
   6814 archer     20   0 1131G  246M  116M S  0.7  1.5  1:09.19 /opt/google/chrome/chrome --type=renderer --c
  25683 systemd-o  20   0 14832  6912  6144 S  0.7  0.0  1:01.71 /lib/systemd/systemd-oomd
      1 root       20   0  162M 11776  8192 S  0.0  0.1  0:04.40 /lib/systemd/systemd --system --deserialize 4
    220 root       19  -1 80480 39808 38528 S  0.0  0.2  0:02.34 /lib/systemd/systemd-journald
    271 root       20   0 26892  6900  4596 S  0.0  0.0  0:00.46 /lib/systemd/systemd-udevd
    489 systemd-r  20   0 26252 14332  9472 S  0.0  0.1  0:04.70 /lib/systemd/systemd-resolved
    490 systemd-t  20   0 89380  7424  6656 S  0.0  0.0  0:00.27 /lib/systemd/systemd-timesyncd
    532 systemd-t  20   0 89380  7424  6656 S  0.0  0.0  0:00.00 /lib/systemd/systemd-timesyncd
    600 root       20   0  234M  7604  6708 S  0.0  0.0  0:02.69 /usr/libexec/accounts-daemon
F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

Process 1412 is running Xorg, let's see what that is.

According to google, X.Org Server is the free and open-source implementation of the X Window System display server stewarded by the X.Org Foundation.

Hmm, so it's my computer's “display server”. What is that?



what is a display server

About 1,370,000,000 results (0.31 seconds)

A display server or window server is a program whose primary task is to coordinate the input and output of its clients to and from the rest of the operating system, the hardware, and each other.

Wikipedia
https://en.wikipedia.org › wiki › Windowing_system

Windowing system - Wikipedia

Awesome, so if I want to do IO stuff, I might want to use Xorg instead of straight system calls. Also if I’m having IO issues, maybe I should check out Xorg and Xorgs logs. Good to know!

Supplementary Material

Here are some really good things to know about. You will probably run into most of these just by doing the above steps.

- Bootloaders (Grub)
- Init systems (systemd, sysV)
- Daemons (continually running background processes - conventionally have a name ending with d (example systemd))
- Window manager (X: Framework for GUI’s)
- GUI (Gnome, wayland, i3) (Are built on top of a window managers like X)