

Linux Primer

Basics

Objectives

- Get around Linux
- Use basic file commands

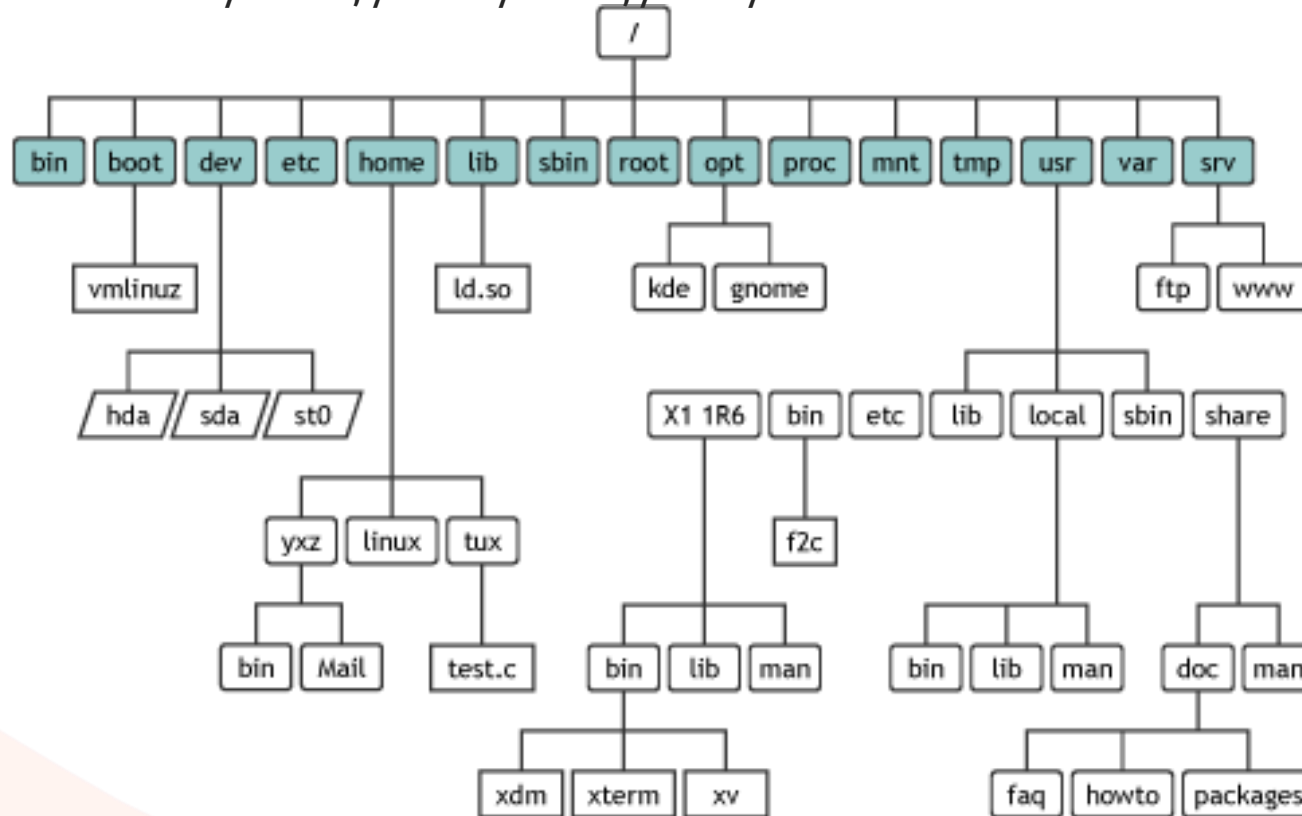
Logging on & Who are you

- Using an SSH Client we will log on to Linux
- Who are you logged in as?

```
ssh me@somelinuxsystem.com
```

Logging on & Who are you

- The Linux file system in short a quick tour
 - /tmp, /home, /usr, /var what are they?
 - Names for directories like /home, /home/steve, / and /root



Getting Help

- Man pages are built into linux so you can use the man command
 - `man ls`
 - `man cp`
- Stackoverflow.com
 - Some useful answers, but not always the simplest, or correct so you should try them out
 - When trying out destructive commands make sure you create a directory with some test files in
- When searching the web, also include the operating system type of RHEL or Debian

```
man ls  
man cp
```

Changing Directory

- Using CD to get about and some useful short cuts
 - Absolute and relative paths
 - tilde (~) and minus (-)

```
# All go home!  
cd ~  
cd  
cd $HOME  
  
# change into specific folders  
cd /home/steve  
cd /etc/bin
```

Creating directories

- Creating directories with mkdir
- Creating child directories when the parent doesn't exist

```
mkdir newfolder  
cd newfolder  
pwd
```

TASK

1. Search the web to find out how to create child directories using mkdir if the parent does not exist
2. Once you have found the command create the directory structure using only one command
 - `xian/projects/frontend`

Viewing File Attributes

- Viewing the attributes of files and directories with `ls`
 - Core options
 - `-l` `-a` `-ltr` `-S` `-R`
- What do all those columns mean?

```
ls somefolder
ls -a somefolder
ls -ltr somefolder
ls -S somefolder
ls -R somefolder
```

TASK

1. List attributes of all files and directories in `/etc/cron.daily`
2. List the file/directories in `/etc/cron.daily` in ascending order of size
3. Show all hidden files in your home directory
4. How do I change the output of the `-l` option to show a more readable file size? e.g. MB, GB, etc. List the `/etc/cron.daily` directory using the option you find.

Command Syntax

- All commands are lowercase
- Spacing is important
 - Between command and options
 - Between options and arguments
 - Between all arguments

command [-options] [arg1 arg2 argn]

spacing!

Four blue arrows originate from the word 'spacing!' and point to the four spaces in the command syntax: the space between 'command' and '[-options]', the space between '[-options]' and '[arg1', the space between 'arg1' and 'arg2', and the space between 'arg2' and 'argn'.

Labs

- Labs will be available on this website:
 - <https://github.com/Spiff77/xian>
- Practice with lab 1

Linux Primer

Working with Files

cat — Concatenate and Display Files

- cat is a command-line tool used to **read, concatenate, and display** the contents of files. We use mostly for reading:

```
echo hello > somefile.txt  
cat somefile.txt  
hello
```

Examples - Capturing Output

- Capturing output using **>**
 - This will create the file if it does not exist
 - This will empty any files that exists

```
echo hello > somefile.txt  
cat somefile.txt  
hello
```

Examples - Appending Output

- Capturing output with **>>**
 - This will create the file if it does not exist
 - This will add to the end of the file (append) if it does exist

```
echo hello > somefile.txt
cat somefile.txt
hello
echo goodbye >> somefile.txt
cat somefile.txt
hello
goodbye
```


File Manipulation

- 4 commands
 - Copy = **cp**
 - Rename/Move = **mv**
 - Delete = **rm**
 - Delete Directory = **rmdir** or **rm -r**
- Syntax of cp and mv
 - **command *src dest***

```
echo hello > somefile.txt
cp somefile.txt otherfile.txt
cat otherfile.txt
hello

rm otherfile.txt
ls
somefile.txt
```

Labs

- Practice with lab 2

Viewing File Content

- Commands to use
 - **cat**
 - **less**
 - space = pg dn, b = pg up, /searchptn, n = next search ptn, N = previous search ptn, q = quit
 - **head**
 - **tail**
 - **tr**

Labs

- Practice with lab 3

Linux Primer

Processes

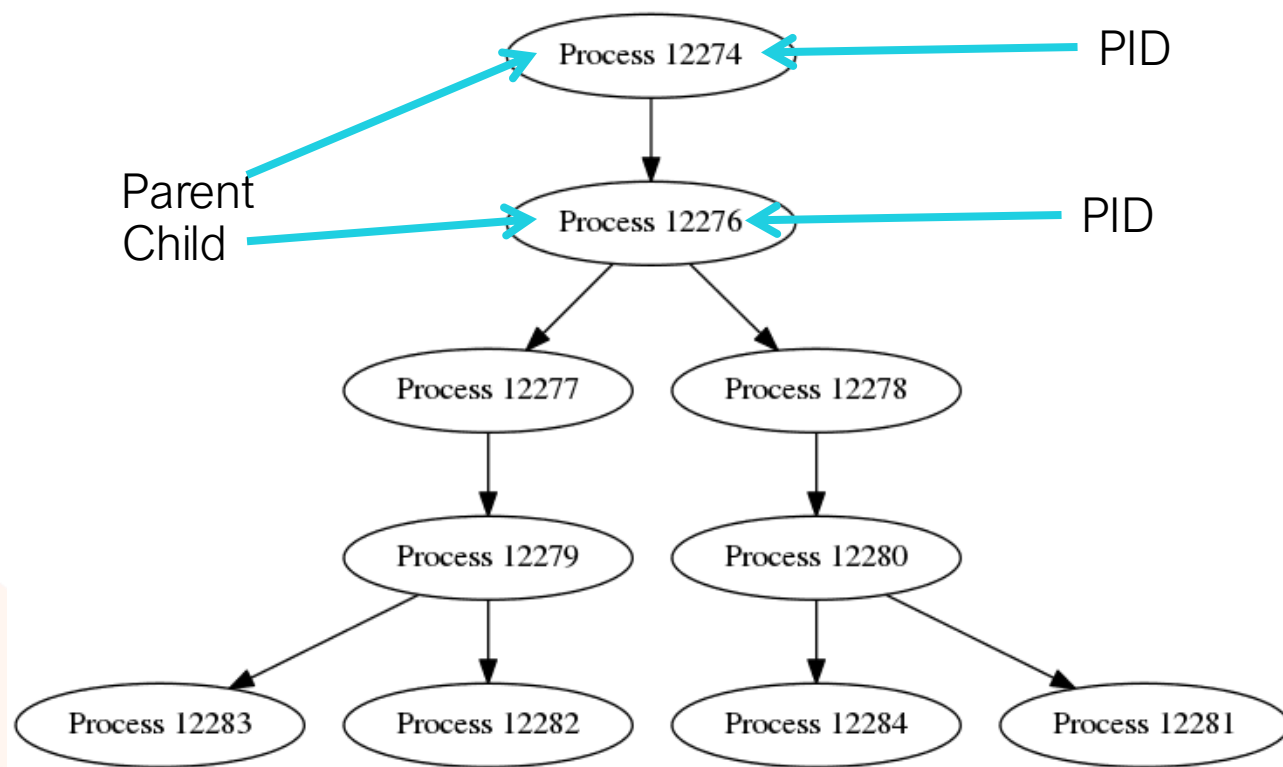
Objectives

- What is a Process
- The Process Tree
- Managing Processes

What is a Process

- All process start as a file containing executable code called a binary
- When we type in the command and press enter it is allocated memory, resource and CPU time
- The PS command to show process status
 - your own processes
 - all processes with extra attributes and what the columns mean
 - getting other attributes using -o

The Process Tree



PS commands

- `ps -f` → Full Format Listing
- `ps -u <user>` → Filter by User
- `ps -forest` → Tree View

```
UID      PID    PPID  CMD
student 3100   3000  python3 -c import subprocess, time; ...
student 3101   3100  \_ python3 -c import time, subprocess; ...
student 3102   3101  \_ sleep 999
student 3000   2800  -bash
```

Managing Processes

- We can signal processes with the kill command or some special CONTROL key presses
 - **CTRL+C** = terminate currently running process on my terminal
 - **kill** PID = terminate the process with the number PID - NOTE only your processes
 - **kill -9** PID = terminate the process regardless - can be dangerous
- A signal informs the process to do something whilst it is running
- Default it to terminate and free up all resources

Labs

- Practice with lab 4