

Going around your machine and commands
30-October-2019

1. Finding your way around the file system

/bin/pwd	The command prompt can be set to not indicate the current directory. This command will print out which directory you are currently in.
cd <location>	Explore shortcuts for locations eg., . for current directory, .. for parent directory, !\$ for argument of previous command, ~ for \$HOME, ~<user> for <user>'s \$HOME etc.
/bin/ls	Explore different options of listing files and directories and the meaning of each of the fields shown.
/bin/dir	Same as above.
/bin/ls <options>	File listing – Permissions, owner, group, size, modified time, name to view the details of a file.
/usr/bin/du	Size of a directory. Explore options for human readable display of output.
/bin/chmod, /bin/chown, /bin/chgrp, /usr/bin/touch, /bin/su, /usr/bin/sudo	Change the permissions, ownership and timestamp of a file. Use su and sudo to perform these operations as administrator or another user.
/usr/bin/file	What type of a file it is? This helps when the file extension is misleading.
/bin/cp, /bin/mv, /bin/mkdir, /bin/rm, /bin/rmdir	Creating, moving and removing files and directories. Explore options to do these recursively, with/without warning etc., Use with care – default execution in Linux is to be silent.
I/O redirection using >, <, >>, , tee <file>	To do things like sending output of a command to another command as input, writing the output of a command to a new file, appending the output of a command to an existing file, reading input for a command from an existing file etc.,
cat, head, tail, more, less	Commands to process I/O and trimming the output horizontally
cut	Trimming the output vertically

grep	Search for a pattern in the input or file
Filters: sort, uniq	Further processing of output.
wc	Count words and lines – useful to count the number of items in the command line output.

Homework:

- [1] Use “file” command in the /bin directory and see what type of executables are out there.
- [2] Count the number of library files that have their names ending with “.so.0” in /usr/lib directory.
- [3] Count the number of shell scripts in /bin directory.
- [4] Explore /var/log folder and see which file is for what purpose. Look at the latest 20 lines of “syslog” and write down what details of system activity is being logged.
- [5] Estimate how much storage is occupied by your /usr directory using a command.
- [6] Go to /usr/share/fonts/truetype folder. Using the recursive option of ls command, find out how many truetype fonts (ending with ttf) are out there in your system.

6. Important locations in the file system

/	Root directory - all the files are below this.
/boot	Boot images are stored in this directory. Often mounted separately as a readonly partition.
/etc	Configuration files. Explore /etc/init directories for programs that start at the time of booting.
/bin and /usr/bin	Binary executables for users
/sbin and /usr/sbin	Binary executables for administrators
/usr/share and /usr/src	Shared and architecture independent stuff across programs including source files
/usr/lib	Runtime library files, particularly important for program development
/usr/local	A place to keep self compiled and third party programs
/var	Run time files of programs including logs
/lib	System libraries
/dev	Device files
/tmp	Temporary directory for the system
/proc	File system to explore processes and system information
/media	Mount locations for removable devices
/mnt	Locally mounted storages
Symbolic Links	Using the command ln one can place a pointer to an alternate location. Particularly useful when storage limitations come up.

Homework:

- [1] When the machine is booting, there is a boot menu shown. The items listed in the boot menu are part of so-called grub menu. Which file has this configuration? Convince yourself that the boot menu shown for your machine is same as what is listed in the configuration file.
- [2] Which file has the name of the host (the name of your machine) written down?
- [3] There is so much of information that the “man” command shows us about commands. Where is all this information stored in the system?
- [4] When you compile a c program, you often use “-lm” to link the math library. This library file is typically named “libm.so.X” where X is a number. Where is this file located in your system?
- [5] Type “who” to see which terminal you are logged in. The terminal numbers start with tty. Look at the file in /dev/ folder with same name and the owner of that file. What does that mean?
- [6] The file “/dev/null” is a special file. What does it contain?
- [7] In the shell, use the “ps” command to identify the process id of your shell. Change directory to “/proc/<pid>” where <pid> is the process id number. This folder contains a bunch of files. Look at the files “status”, “environ” etc., What does this tell you about the process?
- [8] Stick a USB into your machine. Identify which folder the drive is mounted. Find out which file contains the information on where to mount such devices.
- [9] There are soft links (made using “ln -s” command) and hard links (made with “ln” command). Create a file xaa with some command like “ls > xaa”. Create a hard link to xaa by name xab. Using the “-i” option for the command “ls”, check the inode numbers of these two files. Remove the original file xaa. Does this remove the hard link? What does this teach you about hard links? In what are soft links to files different?