

NAME : RAJ PATIL  
ROLL NO: CS18BTECH11039

## HOMEWORK ASSIGNMENT 0

### PLAGIARISM STATEMENT

I certify that this assignment/report is my own work, based on my personal study and/or research and that I have acknowledged all material and sources used in its preparation, whether they be books, articles, reports, lecture notes, and any other kind of document, electronic or personal communication. I also certify that this assignment/report has not previously been submitted for assessment in any other course, except where specific permission has been granted from all course instructors involved, or at any other time in this course, and that I have not copied in part or whole or otherwise plagiarised the work of other students and/or persons. I pledge to uphold the principles of honesty and responsibility at CSE@IITH. In addition, I understand my responsibility to report honour violations by other students if I become aware of it.

Name: Raj Patil  
Date: Sat Feb 15 08:39:04 IST 2020  
Signature: RDP

## Chapter 10: File systems

This chapter was all about the filesystem architecture of linux, its hierarchy, management(Backup and compression) and the different file types.

- The tree starts with the root directory ( maybe referred to as trunk and notified by /). Partitions exist to maintain modularity in the hierarchy based on utility, ownership and type of the data being maintained by that branch of the tree. Filesystems can be mounted on the main tree via a mount point. This can be automated using the fstab file located in the etc directory in root.

- Then, NFS(network file system) is introduced as a way to share data through network systems.

- Then the notion of pseudo filesystems is introduced ( they don't actually exist on the disk and only loaded and maintained onto the main memory ).

- /proc was introduced : to get a feel of its temporary and dynamic nature, I checked the sched file under my current BASH pid ( existed as a folder in /proc ) . Then created a new process and killed the killable using kill -9 -1 and the folder wasn't there anymore but the new one was.

- Then other branches of the root directory such as /var and /boot were introduced.

- diff and patch was introduced in the next section. Also tested diff3 and recursively patched some directories.

- They also clarified the idea that file extensions do not completely signify the characteristics of a file and one should use the file command to avoid disambiguities.

- This was followed by discussion of compression facilities provided and their comparison in terms of use cases and performance. ( gzip,bzip2,xz,zip,tar)
- Then, the nature of dd ( disk to disk ) was discussed ( copying bytes directly ) and how it can be used to make large exact copies of varying file sizes efficiently.

## Chapter 11: Text editors

This one talks about the basic editors ( nano and gedit ) and the advanced ones ( vi and emacs ).

I have already been using vim since 6 months so not elaborating a lot on what they covered in here as it was limited to basic navigation, manipulation and stating the fact that one gets efficient at editing with vim/emacs after they've traversed the steep learning curve : I can attest to that.

That's it for this chapter.

## Chapter 12: User Environment

This chapter talks about the user environment, initially introducing ownership of files in terms of users and groups. Later on it discusses the manipulation of environment variables which can be viewed using commands like set and env. Generally speaking, we can customize the utility being used by us (vim,bash,tmux and so on) by editing the "run config" files (stored as ".\*rc" files, generally in the home directory of the user, otherwise the global files are used) which are referred to every time a new instance of the utility is launched.

Basic commands such as who and whoami are presented to check the current status of user configuration on the system presently

We were also shown some important environment variables, one of them being PATH which we can edit to add a new directory to search for a specific executable being called in a certain order.

The chapter also introduces various keyboard shortcuts for controlling the experience on the terminal, but learning multiple keyboard shortcuts is wasteful and being a user of vim, I've just set my input mode to vi (set editingmode vi) in my .inputrc run config file to be able to use vim keybindings in my terminal and enjoy the convenience without learning multiple shortcuts for the same functionality dependent on the utility being used.

The chapter also displays the use .bash\_historyprofile file that stores the past commands executed in the bash and is only updated once an instance is killed off. File ownership and its manipulation using chown, chgrp is shown and permissions can be modified using chmod.

## Chapter 13: Manipulating Text

This one was all about manipulating text files :

- display and append using cat and echo(using redirection)
- Edit and print using sed and awk
- sorting output and sieving out duplicates using uniq or sort -u

- usage of paste, join and split according to the current use-case
- search for patterns using grep: unix supports three kinds :
  - basic
  - extended
  - Perl-flavoured
  - \* I prefer the extended version
- tr for translating characters, copying stdin to stdout and handling special characters
- other numerous commands such as tee, wc, strings(useful in case of binary files) were introduced as well
- I found less to be very useful as I'd only known about more till date and now I know less, which is more than more
- further convenience commands are tail and head for a quick view of large files.
- the z-prefixed command family was also introduced to work with zipped files before extraction.

## Chapter 14: Network Operations

This chapter was about the basics of network connections and related utilities.

- Firstly, IP addresses were explained along with the different versions (v4 and v6) and the different classes within them.
- DNS(domain name system)'s usage for converting host names to IP addresses is discussed
- tools such as ipconfig, ip, route and ping are introduced
- a quick experiment one could do is test ping <local host> and ping <some remote host> to get an idea about the difference in speeds of packet transfer. The first one serves as a control and will be very small in value.
- Then, various data transfer commands are introduced : curl(the one I use most), wget and the remaining ones for FTP clients. Usage of ssh for remote logins is also discussed.

## Chapter 15: Shell scripting

This chapter was about the basics of shell scripting. A few of the concepts from the previous chapters were restated for completion and relevance to bash scripting. After this chapter, I can write basic scripts that involve rudimentary file handling, arithmetic and very basic control flow.

Scripts are better than bashing out the same sequence of commands again and again for obvious reasons and are a powerfull tool used for automating a workflow.

The Lab assignments did involve the very basic operations to get one's feet wet with shell scripting, including but not limited to

- the usage of environment variables
- handling parameters for function and the script itself
- input and output redirection
- control flow using if;then;elif;then;else

- using expressions
- using test to output booleans for using if : see man test for all the possibilities

## Chapter 16: More Scripting

This one was an extension of the previous chapter and was quite interesting as some info about the internals of the kernel were also mentioned such as random number generation. The rest was the introduction of basic iteration and control flow utilities (syntax) with the underlying minor details.

- The chapter started out with string manipulation : comparison, sorting and getting the length of a string
  - Boolean expressions and operations on them
  - Then the case statement was introduced along with its advantages over the normal if then elif then else paradigm.
  - basic file descriptors ( 0 1 2 ) ( stdin stdout stderr ) were introduced and redirection examples included the usage of /dev/null (black hole) to dump unnecessary outputs
  - Script debugging was also introduced in two ways:
    - invoking bash with a parameter -x to debug the file line by line
    - encapsulating the suspicious section in set -x(turn debugging on) and set +x(turn debugging off).
  - creation of temporaries was also covered (during the execution of a script) and why does one need to randomize the name was emphasized upon. The utility mktemp was introduced and strong use cases for the preference were provided (instead of hard-coded predictable names for temporaries)
  - Lastly, random number generation was covered and the notion of the source of entropy either being derived from hardware (converts noise) or software(used by most modern machines) was introduced and the utilities /dev/random and /dev/urandom were stated : random being the more sophisticated version and urandom being used when performance is a priority.

## Chapter 17: Printing

This one is all about printing and the related utilities and a bit of background info.

- Initially, they cover how to configure a printer ( local or over a network )
- Then, the actual printing process is mentioned
- lastly, manipulation of postscript and PDF files using command line utilities was discussed
  - lots of utilities such as pdftk, pdfinfo, flpsed, pdfmod, enscript, lp, lpr were introduced : not going into the technicalities .
  - CUPS (Common Unix Printing System) is discussed : provides 2 command line utilities: System V and BSD

## Chapter 18: Local Security Principles

This one was about the basic security principles one needs to follow for reducing the risk of an attack and the basic ways in which linux makes it difficult for attackers to do so.

- differences between sudo and su were discussed: always advisable to use sudo : allows for more configuration of user priveleges than su.
- sudo also has the capability log failed attempts to access the utility which can't be done with su.
- the SHA-512 algorithm is stated as the one generally used to encode passwords in the /etc/shadow file
- Pluggable Authentication Modules (PAM) can be configured to check if the passwords created using passwd are strong (the notion of a strong password is also configurable)