Lab 1 : Working with Linux

August 2, 2018

Objective :

- Lab 1 is intended to provide an introduction to Linux. The lab consists of performing basic system operations such as file management, text editing, permission management. The objective of this lab is to make students familiar with the Linux command-line environment and develop the skills of shell scripting. This lab serves as a platform for the subsequent labs related to system calls, process management, file management and memory management.

Recommended Systems/Software Requirements:

- Any flavour of Linux

References:

1. *Unix concepts and applications*, Fourth Edition, Sumitabha Das, TMH.
2. *Unix and shell Programming*, B.A. Forouzan & R.F. Giberg, Thomson.
3. *Beginning shell scripting*, E. Foster, Johnson & other, Wile Y- India

Tutorial

- Tut 1 : Login to the system, open the *Terminal* and type the following on the login prompt :
    1. *echo hello world* :

       The word **hello world** would get displayed as output on the terminal. The *echo* command displays a line of text. type *man echo* to read what an echo command does.
    2. Linux provides on-line manuals for different commands through an interface called **man**
       To know about *man* type the following on the terminal: *man man* and read the description that is displayed.
       For all commands we will use the syntax *man* < *commandname* >
    3. Type *echo $SHELL* : This prints **/bin/bash**
       bash is the name of the login shell that is currently in use

4. Study the following Unix/Linux general purpose utility commands using their *man* pages and execute them on the terminal :
   man, who,cat, cd, cp, ps, ls, mv,rm, mkdir, rmdir, echo, more, date, time, kill, history, chmod, chown, finger, pwd, cal, logout, shutdown

- Tut 2 : Study the following Linux commands : *sed, grep and awk*
  - Use the *sed* command to delete the first character and last character in each line of a file.
  - Use the *grep* command to find how many lines of a file contain a given word. The filname and the word are provided as inputs.

- Tut 3 : Shell Scripting
  You are advised to refer to the text book on *shell scripting* provided in the **Helpful Resources** section of the OS lab webpage before attempting this tutorial.
  - Write a shell script to display "HELLO WORLD" on the terminal :
    1. Open an editor
    2. Type echo HELLO WORLD
    3. save the file with .sh extension (say *test.sh*)
    4. close the editor
    5. At the terminal, type *sh test.sh*
    6. Expected output at the prompt : HELLO WORLD
  - Refer to the **Helpful Resources** section of the OS lab webpage and execute the scripts provided in *Tutorial_shell_scripting*

Assignments

1. Use the *who* command and redirect the result to a file called **myfile1**. Use the *more* command to see the contents of **myfile1**.
2. Use the *date* and *who* commands in sequence (in one line) such that the output of date will display on the screen and the output of who will be redirected to a file called **myfile2**. Use the *more* command to check the contents of **myfile2**
3. Write a *sed* command that swaps the first and second words in each line in a file.
4. Write a (i) shell script program an (ii) *C* program to display "HELLO WORLD"
   - Compare the running time of both the programs using *time* command
5. Write a shell script that takes a command –line argument and reports on whether it is directory, a file, or something else.
6. Write a shell script that accepts one or more file name as arguments and converts all of them to uppercase, provided they exist in the current directory.
7. Write a shell script that determines the period for which a specified user is working on the system.
8. Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.
9. Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.
10. Write a shell script to perform the following string operations:
    (a) To extract a sub-string from a given string
    (b) To find the length of a given string