COLLEGE OF ENGINEERING CHENGANNUR

CS232

FREE AND OPEN SOURCE SOFTWARE LAB RECORD

Submitted By:

Name: Jeny Susan Rajan

Class: S4 D

Roll No: 28

Reg. No: CHN17CS058

CONTENTS

Experiment 1

Linux Commands

Date: 11-02-2019

Aim:

To become familiar with the various basic commands in the Linux operating system. Additional information about each command can be obtained by giving the command 'man' before each command in the Linux terminal.

Description:

The basic commands in Linux are:

- ls : displays all the contents in the directory
  ls –l : displays all the contents in the directory in long list format
  ls –lt : sorts contents in order of time
  ls –ltr : reverses contents to find recently modified files
- mkdir : it allows the user to create directories or folders
- cd : used to change the current directory
- grep : it searches a substring in a file
- cat : used to display the contents of a file, copy text files to a new document, etc.
- rmdir : used to remove an existing directory
- rm : used to remove existing files from your Linux operating system
- pwd : shows which directory you are in (it gives the path from the root)
- clear : used to clear the terminal screen
- cp : used for copying files or group of files or directories
- uname : gives basic information about the computer's hardware and software
- join : used to combine lines of two files on a common field
- adduser : used to create a new user or update default new user information

Result:

Familiarized the basic commands in Linux and their usage.

Experiment 2

Scripting Task

Date: 10-03-2019

Aim:

To calculate the SGPA of first semester and second semester of class S4 D and the CGPA of both semesters and then display the result along with name and register number using appropriate scripting commands.

Description:

- The pdf files of results of both S1 & S2 of the college are downloaded and then converted to text file using: `pdftotext –nopgbrk result_CHN.pdf`
- The results of students in CS are obtained using the grep command: `grep --no-group-separator -A3 'CHN17CS' result_CHN.txt | tr '\n' ' ' | sed 's/\ CHN/\nCHN/g' > results.txt`
- The grades are replaced with grade points using the sed command.
  O, A+, A, B+, B, C , P, F, FE, I
  10, 9, 8.5, 8, 7, 6, 5, 0, 0, 0
- Then SGPA of first semester is calculated using awk command by finding the average of a column in a file. Similarly SGPA of second semester in another file. Then the SGPAs are copied to another file.
- Then the CGPA is calculated with the provided data.
- The resulting file displays:
  Reg No.   Name   S1 SGPA   S2 SGPA   CGPA
- The commands that are used in this task are:
  pdftotext, grep, sed, awk, echo, paste, cat

Result:

Computed the SGPAs in S1 & S2 and obtained the CGPA of the students of S4 D.

Experiment 3

Networking Task

Date: 18-03-2019

Aim:

To set up a network using a network hub.

Description:

A hub, also called a network hub, is a common connection point for devices in a network. Hubs are devices commonly used to connect segments of a LAN.
The hub contains multiple ports. When a packet arrives at one port, it is copied to the other ports so that all segments of the LAN can see all packets.

A network hub is used to set up a connection and establish an ip.

The following commands are used to carry out this task:

- ifconfig : stands for "Interface Configuration". It is a utility for Linux machines to configure, assign, add, delete, control and query network interface in the Linux machine. It is used to assign ip address and netmask to an interface or to disable or enable a given interface.
- ifconfig eth0 : to view the network settings of an adapter
- ifconfig eth0 192.168.0.5 : Assign an ip to an interface eth0
- ifconfig eth0 up : Enables an interface
- ifconfig eth0 down : Disables an interface
- ping : PING (Packet INternet Groper) command is used to test connectivity between two nodes. It is used to verify that a device can communicate with another on a network.

Result:

Multiple devices were connected to a hub. Hubs repeated all of the network traffic sent to them. Hence connection was established between the devices, thereby a network was setup.

Experiment 4

SSH, RSYNC, SCP

Date: 25-03-2019

Aim:

To familiarize with the various data transfer tools and to implement them.

Description:

SSH - ssh stands for "Secure Shell". It is a protocol used to securely connect to a remote server/system. ssh is secure in the sense that it transfers the data in encrypted form between the host and the client. The ssh command provides a secure encrypted connection between two hosts over an insecure network. This connection can also be used for terminal access, file transfers, and for tunneling other applications.

RSYNC – rsync stands for 'remote synchronization'. It is a most commonly used command for copying and synchronizing files and directories remotely as well as locally in Linux/Unix systems.

SCP - stands for secure copy command is used to copy files/folders between servers in secure way. scp basically reads the source file and writes it to the destination. It performs a plain linear copy, locally, or over a network.

The following commands were executed on the terminal:

- ssh user_name@host(ip/domain_name) : used to connect to a remote server or system
- scp source_file destination_file : used to copy files
- rsync -avzp /home/s4d1/b.txt cs17d28@192.168.0.30: : to copy/sync files and directories locally
- rsync -zvp /home/s4d1/b.txt /home/s4d1/Desktop/a.txt : to copy/sync files and directories on a local computer

Result:

With the help of the above commands, files were transferred locally and to the server.

Experiment 5

FTP Usage

Date: 25-03-2019

Aim:

To login to an ftp server with the provided username and password and perform the basic data transfer operations.

Description:

FTP is the simplest file transfer protocol to exchange files to and from a remote computer or network. It implements a basic FTP client that can send, receive, list, delete files and create directories. SFTP(secure file transfer protocol) is an encrypted version of FTP. It transfers files securely over a network connection. It runs over SSH protocol to establish a secure connection.

- sftp remote_username@server_ip or hostname - To open an sftp connection to a remote system.
- sftp> get filename.txt - To download a file from the remote server
- sftp> put filename.txt - To upload a file from a local directory to a remote FTP server
- sftp> get -r remote_directory - To download a directory from the remote system
- sftp> pwd - To check the current working directory
- sftp> ls - To list the files and directories in the remote system
- sftp> bye – to exit the connection

Result:

With the help of the above commands files were uploaded into the remote FTP server .

Experiment 6

Linux Installation

Date: 18-03-2019

Aim:

To install a Linux operating system openSUSE from the provided DVD.

Description:

1. Insert the openSUSE Leap DVD into the drive, then reboot the computer to start the installation program.
2. Select *Installation* on the boot screen, then press ENTER. This boots the system and loads the openSUSE Leap installer. Read the License Agreement. It is presented in the language you have chosen on the boot screen. Proceed with *Next*.
3. A system analysis is performed, where the installer probes for storage devices, and tries to find other installed systems. When the analysis has finished, the *Installation Options* screen launches.
4. Review the partition setup proposed by the system. If necessary, change it.
5. If you have selected *Add Online Repositories Before Installation* earlier during the installation, the *List of Online Repositories* screen lets you choose which official openSUSE Leap online repositories to add.
6. Use the *Installation Settings* screen to review and—if necessary—change several proposed installation settings. The current configuration is listed for each setting.
7. After you have finalized the system configuration on the *Installation Settings* screen, click *Install*. Depending on your software selection you may need to agree to license agreements before the installation confirmation screen pops up. Up to this point no changes have been made to your system. After you click *Install* a second time, the installation process starts.
8. After the installation routine has finished, the computer is rebooted into the installed system.

Result:

openSUSE was successfully installed.

Experiment 7

HTTP and FTP servers

Date: 11-04-2019

Aim:

To set up http and ftp servers.

Description:

A web server is server software, or hardware dedicated to running said software, that can satisfy World Wide Web client requests.

HTTP Server - An HTTP server serves data to clients using the HTTP protocol. It waits for HTTP requests from clients (like your browser) on the Internet and replies with appropriate HTTP responses based on the given HTTP requests.

FTP Server - An FTP Server is a piece of software that is running on a computer and uses the File Transfer Protocol to store and share files. It has a file transfer protocol (FTP) address and is dedicated to receiving an FTP connection. An FTP server needs a TCP/IP network for functioning and is dependent on usage of dedicated servers with one or more FTP clients.

The following commands were executed to setup the servers:

- sudo apt-get install nginx - To install the nginx web server
- sudo /usr/sbin/nginx - To start the server
- sudo gedit /etc/nginx/sites-enabled-default - Update the configuration files
- sudo gedit /var/www/html/index.html - To edit the index.html file
- sudo apt-get install vsftpd - To install vsftpd FTP server
- sudo nano /etc/vsftpd.conf - To configure FTP access
- ftp -p 192.168.0.189 - To test FTP Access

Result:

HTTP and FTP servers are set up.

Experiment 8

(a)Package Management

Date: 24-04-2019

Aim:

To install consistent sets of binary packages to the system from the archive.

Description:

Package management is a method of installing and maintaining (which includes updating and probably removing as well) software on the system. The Debian package management system, based on a tool called dpkg with the very popular apt system, is a powerful, popular, and useful method of package management.

The following commands are used:

- sudo apt update - It downloads the package lists from the repositories and "updates" them to get information on the newest versions of packages and their dependencies.
- sudo apt upgrade - It will fetch new versions of packages existing on the machine
- sudo apt search <packagename> - It searches for the package to see if its available or not
- sudo apt install <packagename> - To install the package
- sudo apt show <packagename>
- sudo apt full-upgrade
- sudo apt remove <packagename> - To uninstall the package
- sudo apt purge <packagename> - It uninstalls the package along with the configuration files.

Result:

The package was installed successfully and later uninstalled with remove and purge.

(b) Perl

Date: 09-05-2019

Aim:

To install perl and run the perlscript (perl program).

Description:

Perl stands for "Practical Extraction and Reporting Language". Perl is a programming language that can be used to perform tasks that would be difficult or cumbersome on the command line. Perl is included by default with most GNU/Linux distributions. Usually, one invokes Perl by using a text editor to write a file and then passing it to the perl program. Perl is a family of scripting programming languages. Perl includes no. of popular UNIX facilities such as sed, awk, tr etc. Perl is a family of two high-level, general-purpose, interpreted, dynamic programming languages.

The following commands are used:

- perl –v – To check if perl is already installed
- sudo apt-get install perl - To install perl
- touch perlprgrm.pl - Create a file named perlpg.pl
- cat >perlprgrm.pl – to write the perlscript
- perl perlprgrm.pl – to run the perlscript

Result:

Following the steps mentioned above perl was successfully installed and the script was run.

(c) LAMP stack

Date: 09-05-2019

Aim:

To install and configure LAMP Stack.

Description:

LAMP is an open source Web development platform that uses Linux as the operating system, Apache as the Web server, MySQL as the relational database management system and PHP as the object-oriented scripting language. (Sometimes Perl or Python is used instead of PHP.)

The Apache web server is among the most popular web servers in the world. It's well-documented and has been in wide use for much of the history of the web, which makes it a great default choice for hosting a website.

- sudo apt install apache2 - To install apache using ubuntu package manager

MySQL is a database management system. Basically, it will organize and provide access to databases where your site can store information.

- sudo apt install mysql-server - To install mysql
- sudo mysql_secure_installation - To run the security script

PHP Hypertext Preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML.

- sudo apt install php libapache2-mod-php php-mysql - To install php and related packages
- sudo systemctl restart apache2 - To restart the apache web server
- sudo gedit /var/www/html/info.php - To create a basic PHP script

Result:

Apache, MySQL and PHP were installed and a sample PHP script was run.

(d)Kernel Compilation

Date: 27-04-2019

Aim:

To build(compile) and install the latest linux kernel from source.

Description:

The various steps involved in building and installing a kernel are as follows:

- wget https://cdn.kernel.org/pub/linux/kernel/v4.x/linux-4.20.12.tar.xz - To get the latest Linux kernel source code
- unzx -v linux-4.20.12.tar.xzb - Extract tar.xz file
- cd linux-4.20.12
- cp -v /boot/config-$(uname -r) .config - Configure the Linux kernel features and modules
- sudo apt-get install build-essential libncurses-dev bison flex libssl-dev libelf-dev - To install the required compilers and other tools
- sudo make modules_install - To install the Linux kernel modules
- sudo make install - To install the linux kernel
- sudo update-grub - Update the grub config

Result:

The latest Linux kernel was successfully build and installed.

Experiment 9

Webpage Creation on Server

Date: 09-05-2019

Aim:

To create own webpage on server. Upload contents from your home directory to 192.168.0.30. The pages could be accessed by http://192.168.0.30/cs17d/cs17d28/ .

Description:

Webpage is created using HTML and the links to different files are provided along with it.

A table was created which displays the following :

- No.
- Experiment
- Code link (typescript link)
- Date of upload
- Description
- Shell script Link

The webpage is made accessible as:
http://192.168.0.30/cs17d/cs17d28

Result:

The created webpage was successfully uploaded into the FTP server and was made accessible with the relevant files.

Experiment 10

Linux Tutorial pages 7-12

Date: 10-05-2019

Aim:

To familiarize with the Linux commands provided in the tutorial in slides 7 to 12 .

Description:

○ Directory and File handling

- mkdir : mkdir command in Linux allows the user to create directories (also referred to as folders in some operating systems). This command can create multiple directories at once as well as set the permissions for the directories.
  Syntax: mkdir directory_name
- cd : The cd command, also known as chdir (change directory), is a command-line OS shell command used to change the current working directory in operating system.
  Syntax: cd directory_name
- rmdir : rmdir command is used remove empty directories from the filesystem in Linux. The rmdir command removes each and every directory specified in the command line only if these directories are empty. So if the specified directory has some directories or files in it then this cannot be removed by rmdir command.
  Syntax: rmdir directory_name
- cp : cp stands for copy. This command is used to copy files or group of files or directory. It creates an exact image of a file on a disk with different file name. cp command require at least two filenames in its arguments.
  Syntax: cp Src_file Dest_file #copying 1 file
  cp Src_file1 Src_file2 Src_file3 Dest_directory #copy 2 files to a directory
  cp -R Src_directory Dest_directory #copying 1 directory
- cp -p : it also copies files.
  Syntax: cp -p file1.txt file2.txt

- rm : used remove files or directories

  ```
  Syntax: rm filename                 #removes file
              rm directoryname                #removes directory
  ```

- rm -r : rm prompts the user for whether to proceed with the entire operation. If the response is not affirmative, the entire command is aborted.

  Syntax: rm -r filename

- mv : mv stands for move. mv is used to move one or more files or directories from one place to another in file system like UNIX. It has two distinct functions:

  1) It rename a file or folder.
  2) It moves group of files to different directory.

  No additional space is consumed on a disk during renaming. This command normally works silently means no prompt for confirmation.

  Syntax: mv [Option] source destination

  ```
  eg. $ ls
             a.txt  b.txt  c.txt  d.txt
      $ mv a.txt geek.txt
      $ ls
             b.txt  c.txt  d.txt  geek.txt
  ```

- locate : find files by name. locate reads one or more databases prepared by updatedb and writes file names matching at least one of the PATTERNs to standard output, one per line. The locate utility works better and faster than find command counterpart because instead of searching the file system when a file search is initiated, it would look through a database. This database contains bits and parts of files and their corresponding paths on your system. By default, locate command does not check whether the files found in the database still exist and it never reports files created after the most recent update of the relevant database.

  Syntax: locate filename

- updatedb : locate command relies on a database called mlocate. The said database needs to be updated regularly for the command utility to work efficiently. To update the mlocate database, you use a utility called updatedb. updatedb creates or updates a database used by locate. If the database already exists, its data is reused to avoid rereading directories that have not changed.

- find /-name abc : used to check whether a file is present in a directory.

  Syntax: find/<filename> directoryname

- df : df is a standard Unix command used to display the amount of available disk space for file systems on which the invoking user has appropriate read access.

- du : du command, short for disk usage, is used to estimate file space usage. The du command can be used to track the files and directories which are consuming excessive amount of space on hard disk drive.

○ Remote access

- ssh commands:
  ssh â€” OpenSSH SSH client (remote login program)
  ssh (SSH client) is a program for logging into a remote machine and for executing commands on a remote machine.It is intended to provide secure encrypted communications between two untrusted hosts over an insecure network.
  Syntax: ssh hostname : remotely logins in hostname
  ssh 14.139.189.215 : logins remotely into ip address

- scp commands:
  SCP (secure copy) is a command line utility that allows you to securely copy files and directories between two locations. With scp, you can copy a file or directory: From your local system to a remote system. From a remote system to your local system.
  Syntax:

```
scp filename.txt username@ip_address:
 #copies the file to the remote system
```

- rsync commands:
  Rsync (Remote Sync) is a most commonly used command for copying and synchronizing files and directories remotely as well as locally in Linux/Unix systems.
  Syntax:

```
rsync          #version and usage given
rsync -a       #fast, versatile, remote (and local) file-copying tool
rsync -a file.txt username@ip_address
 #copies file to remote system
rsync -a {file1.txt,file2.txt} directoryname
 #copies the files to the given directory
```

○ Process

- ps: The ps (i.e., process status) command is used to provide information about the currently running processes, including their process identification numbers (PIDs).
- ps -ef : The -e option generates a list of information about every process currently running. The -f option generates a listing that contains fewer items of information for each process than the -l option.

- kill : kill command in Linux (located in /bin/kill), is a built-in command which is used to terminate processes manually. kill command sends a signal to a process which terminates the process.
- pkill - look up or signal processes based on name and other attributes.
- top : top command displays processor activity of your Linux box and also displays tasks managed by kernel in real-time. It'll show processor and memory are being used and other information like running processes. This may help you to take correct action.
- service appache start : service runs a System V init script or systemd unit in as predictable an environment as possible, removing most environment variables and with the current working directory set to /.

○ Redirection

- ll>abc.tct : created a file abc.tct in long list of files and directories.
- ps -ef | grep ask : look for lines containing processname in a detailed overview/snapshot of all current processes, and display those lines.

○ Users

- who : shows who is logged on
- who -a : shows all who are logged on
- whoami : Print the user name associated with the current effective user ID.
- id //user, group id :
- id -u //user id :
- id -un //user name :
- last : shows a listing of last logged in users
- last -n9 : show a listing of 9 last logged in users
- useradd ask : create a new user or update default new user information
- useradd -d /home/456 456 : adduser 456
- usermod: the command 'usermod' is used to modify or change any attributes of a already created user account via command line. The command 'usermod' is similar to that 'useradd' or 'adduser' but the login granted to an existing user.
- passwd ask : The passwd command changes passwords for user accounts.
- deluser : deluser and delgroup remove users and groups from the system according to command line options and configuration information
- su:change user ID or become superuser

○ mount

- mount: mount a filesystem
- umount: unmount file systems

Code:

```
mkdir 123
ls
kernel.txt
 14.139.189.217
 123                                      LAMP.sh
 abcde.txt                                Lamp.txt
 abc.tct
 abc.txt                                  linuxcommands.sh
 add.pl                                   _[01;34mMusic_[0m
 add.sh                                   name.txt
 a.txt                                    networking.sh
 b.txt                                    networking.txt
 B.txt                                    no.txt
 ceccs1711.2.19.txt                       perl.txt
 res.txt
 ceccs17d25.03.2019.txt                   result_CHN2.pdf
 ceccs17d.txt                             result_CHN2.txt
 cgpa                                     result_CHN.pdf
'cgpa calculation.sh'                     result_CHN.txt
 cgpacalculation.txt                      results2.txt
 cgpa.txt                                 resultss1.tx
 results.txt
 C.txt                                    result.txt
 examples.desktop                         s1sgpa.txt
 final_cgpa.txt                           s2final.txt
 fosslab.txt                              s2sgpa.txt
 foss.txt                                 s4dfinaldoc.txt
 ftp.sh                                   s4d.txt
'ftp usage.sh'                            scpsshrsync.sh
 http.sh                                  _[01;34mVideos_[0m
 i.html                                   webpage.html
 index.html                               world.txt
 kernel.sh                                yes.txt

cd 123

_]0;jeny@jenys-workspace: ~/123 _[01;32mjeny@jenys-
workspace_[00m:_[01;34m~/123_[00m$ cd ~/

rmdir 123
cat a.txt
1
2
3
cat b.txt
a
b
c
cp a.txt b.txt

cat b.txt
1
2
3
```

```
cat c.txt
4
5
6
cp -p a.txt c.txt
cat c.txt
1
2
3

rm c.txt
rm -r a.txt

cat yes.txt
world
cat no.txt
hello
mv no.txt yes.txt
cat yes.txt
hello

locate yes.txt
/home/jeny/yes.txt
sudo updatedb
find /yes.txt home
df
Filesystem     1K-blocks    Used Available Use% Mounted on
udev             4005700       0   4005700   0% /dev
tmpfs             805468    2044    803424   1% /run
/dev/sda7       57410400 6837944  47626460  13% /
tmpfs            4027320   90872   3936448   3% /dev/shm
tmpfs               5120       4      5116   1% /run/lock
tmpfs            4027320       0   4027320   0% /sys/fs/cgroup
/dev/loop0         36224   36224         0 100% /snap/gtk-common-themes/1198
/dev/loop4         55040   55040         0 100% /snap/core18/782
/dev/loop6        146944  146944         0 100% /dev/loop24        91648   91648
0 100% /snap/core/6818
/dev/loop21       154752  154752         0 100% /snap/gnome-3-28-1804/36
/dev/sda8       92502680 2461764  85298928   3% /home
/dev/sda5         967320  125984    774984  14% /boot
tmpfs             805464      20    805444   1% /run/user/121
tmpfs             805464      84    805380   1% /run/user/1000
/dev/loop26        15104   15104         0 100% /snap/gnome-characters/258
du
208   ./Desktop/Linux Installation
165540     ./Desktop/Intrnship diaries1
165756     ./Desktop
3524  ./Videos/Webcam
3528  ./Videos
180   ./Pictures/Wallpapers
184   ./Pictures
8     ./.config/ibus/bus
12    ./.config/ibus
16    ./.config/evolution/sources
20    ./.config/evolution
20    ./.config/sublime-text-3/Local
4     ./.config/sublime-text-3/Packages/User
8     ./.config/sublime-text-3/Packages
```

```
28   ./.config/sublime-text-3/Cache/Python
316  ./.config/sublime-text-3/Cache/HTML
28   ./.config/sublime-text-3/Cache/JavaScript
28   ./.config/sublime-text-3/Cache/R
28   ./.config/sublime-text-3/Cache/C#
24   ./.config/sublime-text-3/Cache/Rails
4    ./Music
4    ./.mozilla/extensions
4    ./.mozilla/systemextensionsdev
60   ./.mozilla/firefox/mjo32lrj.default/features

ssh cs17d28@14.139.189.217
.
              |
.  .    . .-. | .-..-. .--.--. .-.
 \  \  / (.-' |(  (   )|  |  |(.-'
  `' `'   `__'` `_'`_' '  '  ` `__'
cs17d28@14.139.189.217's password:
Welcome to elementary OS 0.4.1 Loki (GNU/Linux 4.13.0-32-generic x86_64)

 * Website:  http://elementary.io/

99 packages can be updated.
0 updates are security updates.

Last login: Mon May 10 02:13:41 2019 from 117.230.160.106$ exit
Connection to 14.139.189.217 closed.
scp yes.txt cs17d28@14.139.189.217:

              |
.  .    . .-. | .-..-. .--.--. .-.
 \  \  / (.-' |(  (   )|  |  |(.-'
  `' `'   `__'` `_'`_' '  '  ` `__'
cs17d28@14.139.189.217's password:

yes.txt                                         0%    0     0.0KB/s   --:-- ETA
yes.txt                                       100%    6     0.1KB/s   00:00

rsync
rsync -a
rsync -a a.txt cs17d28@14.139.189.217:
              |
.  .    . .-. | .-..-. .--.--. .-.
 \  \  / (.-' |(  (   )|  |  |(.-'
  `' `'   `__'` `_'`_' '  '  ` `__'
cs17d28@14.139.189.217's password:

rsync -a {yes.txt,a.txt}home
cd home
_]0;jeny@jenys-workspace: ~/home _[01;32mjeny@jenys-
workspace_[00m:_[01;34m~/home_[00m$ ls
a.txt  no.txt  world.txt  yes.txt
cd ~/

ps
  PID TTY          TIME CMD
 6138 pts/2    00:00:00 bash
 6260 pts/2    00:00:00 ps
```

```
ps -ef
UID        PID  PPID  C STIME TTY          TIME CMD
root         1     0  0 05:27 ?        00:00:13 /sbin/init splash
root         2     0  0 05:27 ?        00:00:00 [kthreadd]
root         4     2  0 05:27 ?        00:00:00 [kworker/0:0H]
root         6     2  0 05:27 ?        00:00:00 [mm_percpu_wq]
root         7     2  0 05:27 ?        00:00:00 [ksoftirqd/0]
root         8     2  0 05:27 ?        00:00:08 [rcu_sched]
root         9     2  0 05:27 ?        00:00:00 [rcu_bh]
root        10     2  0 05:27 ?        00:00:00 [migration/0]
root        11     2  0 05:27 ?        00:00:00 [watchdog/0]
root        12     2  0 05:27 ?        00:00:00 [cpuhp/0]
d -q -sf /usr/li
root      5233     2  0 08:26 ?        00:00:00 [kworker/3:1]
jeny      5652  1480  0 08:49 ?        00:00:00 /usr/lib/gvfs/gvfsd-mtp --spawne
root      5815     2  0 08:59 ?        00:00:00 [kworker/u16:3]
root      5921     2  0 09:08 ?        00:00:00 [kworker/u16:1]
root      6029     2  0 09:12 ?        00:00:00 [kworker/2:1]
jeny      6129  2102  0 09:18 pts/0    00:00:00 bash
jeny      6137  6129  0 09:18 pts/0    00:00:00 script foss.txt
jeny      6138  6137  0 09:18 pts/2    00:00:00 bash -i
jeny      6166  2102  0 09:20 pts/1    00:00:00 bash
root      6221     2  0 09:27 ?        00:00:00 [kworker/u16:2]
root      6261     2  0 09:32 ?        00:00:00 [kworker/1:0]
jeny      6262  6138  0 09:32 pts/2    00:00:00 ps -ef

kill
kill: usage: kill [-s sigspec | -n signum | -sigspec] pid | jobspec ... or kill -
l [sigspec]

kill 6138
pkill dhcp
top
service appache start

ll >abc.tct
ll
_[?1h_=_[?25l_[H_[2J_(B_[mtop - 09:34:48 up  4:07,  1 user,  load average: 1.33,
1.42, 1.45_(B_[m_[39;49m_(B_[m_[39;49m_[K
Tasks:_(B_[m_[39;49m_[1m 297 _(B_[m_[39;49mtotal,_(B_[m_[39;49m_[1m   1
_(B_[m_[39;49mrunning,_(B_[m_[39;49m_[1m 242
_(B_[m_[39;49msleeping,_(B_[m_[39;49m_[1m   0
_(B_[m_[39;49mstopped,_(B_[m_[39;49m_[1m   0 total 951820
-rw-r--r--  1 jeny jeny        8 May 13 06:16  abcde.txt
-rw-r--r--  1 jeny jeny     6024 May 13 09:35  abc.tct
-rw-rw-r--  1 jeny jeny      137 Apr 30 20:29  add.pl
-rw-r--r--  1 jeny jeny      163 May 11 16:21  add.sh
-rw-r--r--  1 jeny jeny      360 Apr 30 20:35  ftp.sh
-rw-r--r--  1 jeny jeny      130 May 10 23:48 'ftp usage.sh'
-rw-rw-r--  1 jeny jeny     6828 May 13 06:39  index.html
-rw-r--r--  1 jeny jeny      342 Apr 30 07:17  kernel.sh
-rw-r--r--  1 jeny jeny   191984 Apr 30 07:19  kernel.txt
-rw-r--r--  1 jeny jeny      433 Apr 30 20:37  LAMP.sh
-rw-r--r--  1 jeny jeny      188 Apr 30 18:42  Lamp.txt
-rw-r--r--  1 jeny jeny      108 May 10 07:25  networking.sh
-rw-r--r--  1 jeny jeny     7690 May 10 07:25  networking.txt
-rw-rw-r--  1 jeny jeny    29005 Mar 25 22:10  result_CHN2.pdf
-rw-r--r--  1 jeny jeny    42569 May 10 07:35  result_CHN2.txt
```

```
-rw-r--r--  1 jeny jeny       204 May 10 23:36  scpsshrsync.sh
-rw-r--r--  1 jeny jeny       903 May 10 07:35  sgpacalc.txt
-rw-r--r--  1 jeny jeny       240 May 10 07:36  sgpas2.txt
-rw-r--r--  1 jeny jeny         0 Mar 25 21:44  sgpa.txt
-rw-rw-r--  1 jeny jeny      6726 May 11 15:14  webpage.html

ps -ef | grep ask
root        33     2  0 05:27 ?        00:00:00
[rcu_t_[01;31m_[Kask_[m_[Ks_kthre]
root        37     2  0 05:27 ?        00:00:00 [khungt_[01;31m_[Kask_[m_[Kd]
jeny      6312  6138  0 09:36 pts/2    00:00:00 grep --color=auto
_[01;31m_[Kask_[m_[K

who
jeny     :0           2019-05-13 05:28 (:0)

who -a
           system boot  2019-05-13 05:27
           run-level 5  2019-05-13 05:28
jeny     ? :0           2019-05-13 05:28   ?          1331 (:0)

whoami
jeny

last
jeny     :0           :0               Mon May 13 05:28   still logged in
reboot   system boot  4.15.0-47-generi Mon May 13 05:27   still running
jeny     :0           :0               Sun May 12 16:15 - down   (03:29)
reboot   system boot  4.15.0-47-generi Sun May 12 16:13 - 19:44  (03:31)
reboot   system boot  4.15.0-47-generi Sun May 12 15:24 - 15:25  (00:00)
jeny     :0           :0               Fri May 10 20:59 - 16:40  (19:40)
reboot   system boot  4.15.0-47-generi Fri May 10 20:56 - 16:40  (19:43)
jeny     :0           :0               Thu May  9 19:15 - 20:55 (1+01:40)
reboot   system boot  4.15.0-47-generi Thu May  9 19:14 - 20:56 (1+01:41)

wtmp begins Sat May  4 19:55:03 2019

last -n9
jeny     :0           :0               Mon May 13 05:28   still logged in
reboot   system boot  4.15.0-47-generi Mon May 13 05:27   still running
jeny     :0           :0               Sun May 12 16:15 - down   (03:29)
reboot   system boot  4.15.0-47-generi Sun May 12 16:13 - 19:44  (03:31)
reboot   system boot  4.15.0-47-generi Sun May 12 15:24 - 15:25  (00:00)
jeny     :0           :0               Fri May 10 20:59 - 16:40  (19:40)
reboot   system boot  4.15.0-47-generi Fri May 10 20:56 - 16:40  (19:43)
jeny     :0           :0               Thu May  9 19:15 - 20:55 (1+01:40)
reboot   system boot  4.15.0-47-generi Thu May  9 19:14 - 20:56 (1+01:41)

wtmp begins Sat May  4 19:55:03 2019

last lpp

wtmp begins Sat May  4 19:55:03 2019

sudo useradd 123

useradd -d /home/456 456
```

```
usermod


passwd ask
passwd: You may not view or modify password information for ask.

deluser
/usr/sbin/deluser: Only root may remove a user or group from the system.

sudo su
workspace:/home/jeny# exit
exit

sudo su 123
$ exit

mount /media/usb /home/ask/usb
umount /media/usb

exit
```

Result:

Familiarized with the Linux commands provided in the tutorial and executed the same successfully.