

SOFTSERVE PYTHON DEVELOPER INTERNSHIP

REPORT ON LINUX OPERATING SYSTEM

Report Number 1

“User information lookup program”

Intern name: Bahamboula Idourah C. Y.

City: Kharkov

Group: Kh-064-Python

Instructor: Дмитрий Узун

Kharkov


Goal: Acquire knowledge on looking up user information

Objectives:

- work with Linux operating system in virtual machine
- execute commands and screen shot the results
- report the result of the laboratory work

i. Task 1

1) Log in to system as root (see Fig 1.1)



```
student@CsnKhai:~$ sudo su
[sudo] password for student:
root@CsnKhai:/home/student#
```

Fig 1.1 Log in as root using sudo su command

- 2) Use the passwd command to change the password (see Fig 1.2).
Examine the basic parameters of command (see Fig 1.3.). What system file does it change?

```
Ubuntu [En fonction] - Oracle VM VirtualBox
Fichier Machine Écran Entrée Périphériques Aide
student@CsnKhai:~$ sudo su
[sudo] password for student:
root@CsnKhai:/home/student# passwd
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@CsnKhai:/home/student# _
```

Fig 1.2 changing user password using passwd

```
root@CsnKhai:/home/student# passwd -help
Usage: passwd [options] [LOGIN]

Options:
  -a, --all                report password status on all accounts
  -d, --delete             delete the password for the named account
  -e, --expire             force expire the password for the named account
  -h, --help              display this help message and exit
  -k, --keep-tokens        change password only if expired
  -i, --inactive INACTIVE set password inactive after expiration
                           to INACTIVE
  -l, --lock               lock the password of the named account
  -n, --mindays MIN_DAYS  set minimum number of days before password
                           change to MIN_DAYS
  -q, --quiet              quiet mode
  -r, --repository REPOSITORY change password in REPOSITORY repository
  -R, --root CHROOT_DIR   directory to chroot into
  -S, --status             report password status on the named account
  -u, --unlock             unlock the password of the named account
  -w, --warndays WARN_DAYS set expiration warning days to WARN_DAYS
  -x, --maxdays MAX_DAYS set maximum number of days before password
                           change to MAX_DAYS

root@CsnKhai:/home/student# _
```

Fig 1.3 passwd parameter using help command

Passwd changes /etc/passwd

- 3) Determine the users registered in the system, as well as what commands they execute. What addition information can be gleaned from command execution?

```
root@CsnKhai:/home/student# finger
Login      Name           Tty          Idle   Login Time   Office       Office Phone
student    Student Khai  *tty1                Feb 16 07:14
root@CsnKhai:/home/student#
```

Using finger command, we can see the log in time, the office phone.

4) Change user personal information

```
root@CsnKhai:/home/student# chfn
Changing the user information for root
Enter the new value, or press ENTER for the default
  Full Name [Idourah Christ Yoane]:
  Room Number []: 511
  Work Phone []: +380 66 370 81 94
  Home Phone [380663708194]: +380 66 370 81 94
  Other []: idourah96@gmail.com
root@CsnKhai:/home/student#
```

5) Become familiar with the linux help system and the man inof

```
student@CsnKhai:~$ passwd -S
student P 02/14/2022 0 99999 7 -1
student@CsnKhai:~$ _
```

5.1 passwd command using `-S` key which report passord of the name account

```
student@CsnKhai:~$ sudo passwd -e student
passwd: password expiry information changed.
student@CsnKhai:~$
```

5.2 passwd command using `-e` key to force password to expire

```
student@CsnKhai:~$ sudo chfn -f idourah
student@CsnKhai:~$ sudo chfn -h +380663708194
student@CsnKhai:~$
```

5.3 chfn `-f` and chfn `-h`. `-f` key to change full name and `-h` home phone number

6) Explore the more and less commands using the help system.

SUMMARY OF LESS COMMANDS

Commands marked with * may be preceded by a number, **N**.
Notes in parentheses indicate the behavior if **N** is given.
A key preceded by a caret indicates the Ctrl key; thus ^K is ctrl-K.

h H Display this help.
q :q Q :Q ZZ Exit.

MOVING

e ^E j ^N CR * Forward one line (or **N** lines).
y ^Y k ^K ^P * Backward one line (or **N** lines).
f ^F ^V SPACE * Forward one window (or **N** lines).
b ^B ESC-v * Backward one window (or **N** lines).
z * Forward one window (and set window to **N**).
w * Backward one window (and set window to **N**).
ESC-SPACE * Forward one window, but don't stop at end-of-file.
d ^D * Forward one half-window (and set half-window to **N**).
u ^U * Backward one half-window (and set half-window to **N**).
ESC-) RightArrow * Left one half screen width (or **N** positions).
ESC-(LeftArrow * Right one half screen width (or **N** positions).
F Forward forever; like "tail -f".
r ^R ^L Repaint screen.
R Repaint screen, discarding buffered input.

```
root@CsnKhai:/usr/share/menu# less bash
?package(bash):needs="text" section="Applications/Shells" title="Bash" command="/
/bin/bash --login"
?package(bash):needs="text" section="Applications/Shells" title="Sh" command="/b
in/sh --login"
bash (END)
```

```
root@CsnKhai:/home/student/bash-file# more script.bash
#!/bin/bash
# A small bash script to say helloworld as it is a tradition in modern language
echo "Helloworld writting script using bash is time consuming but funny"
root@CsnKhai:/home/student/bash-file# _
```

```
root@CsnKhai:/usr/share/menu# more bash
?package(bash):needs="text" section="Applications/Shells" title="Bash" command="/
/bin/bash --login"
?package(bash):needs="text" section="Applications/Shells" title="Sh" command="/b
in/sh --login"
root@CsnKhai:/usr/share/menu# _
```

7) List the contents of the home directory using the ls command

```
root@CsnKhai:/home/student/bash-file# ls
script.bash
root@CsnKhai:/home/student/bash-file# less script.bash
#!/bin/bash
# A small bash script to say helloworld as it is a tradition in modern language
echo "Helloworld writting script using bash is time consuming but funny"
script.bash (END)
```

```
root@CsnKhai:/home# ls
student
root@CsnKhai:/home# ls -a
.  ..  student
root@CsnKhai:/home# ls -l --author
total 4
drwxr-xr-x 4 student student student 4096 Feb 16 08:29 student
root@CsnKhai:/home# ls -al
total 12
drwxr-xr-x  3 root    root    4096 Sep 15  2015 .
drwxr-xr-x 21 root    root    4096 Sep 15  2015 ..
drwxr-xr-x  4 student student 4096 Feb 16 08:29 student
root@CsnKhai:/home#
```

Task 2

2.1 Examine the tree command. Master the technique of applying a template, for example, display all files that contain a character c, or files that contain a specific sequence of characters. List subdirectories of the root directory up to and including the second nesting level.

```
root@CsnKhai:/home/student# tree -P fibonacci*
.
├── algorithms
│   ├── fibonacci.c
│   ├── fibonacci.cpp
│   └── fibonacci.py
└── bash-file

2 directories, 3 files
root@CsnKhai:/home/student#
```

Fig 2.1 displaying files containing Fibonacci using tree command

```
├── hypervisor
├── kernel
├── module
├── power
├── tmp
├── usr
│   ├── bin
│   ├── games
│   ├── include
│   ├── lib
│   ├── local
│   ├── sbin
│   ├── share
│   └── src
├── var
│   ├── backups
│   ├── cache
│   ├── lib
│   ├── local
│   ├── lock -> /run/lock
│   ├── log
│   ├── mail
│   ├── opt
│   ├── run -> /run
│   ├── spool
│   └── tmp
└── vmlinuz -> boot/vmlinuz-3.13.0-63-generic
```

Fig 2.1.2 Listing subdirectories in root including the second level

2.2. What command can be used to determine the type of file (for example, text or binary)? Give an example.

```
root@CsnKhai:/home/student/algorithms# file fibonacci.c
fibonacci.c: C source, ASCII text
root@CsnKhai:/home/student/algorithms# _
```

Fig 2.2.1. File type using file command

```
root@CsnKhai:/home/student/algorithms# ls
fibonacci fibonacci.c fibonacci.cpp fibonacci.py hello.py
root@CsnKhai:/home/student/algorithms# file fibonacci
fibonacci: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically linked (uses shared libs), for GNU/Linux 2.6.24, BuildID[sha1]=7c11ea5670dfcbc6105a1a368abe90b2ef431639, not stripped
```

Fig 2.2.2. Showing type of file using file command

2.3. Master the skills of navigating the file system using relative and absolute paths. How can you go back to your home directory from anywhere in the file system?

```
root@CsnKhai:/proc/driver# cd /home
root@CsnKhai:/home# cd student/
root@CsnKhai:/home/student# ls
algorithms  bash-file  class1  homework1  output  text
root@CsnKhai:/home/student# ls algorithms/
fibonacci  fibonacci.c  fibonacci.cpp  fibonacci.py  hello.py
root@CsnKhai:/home/student#
```

Fig 2.3.1.Navigating the file system using relative and absolute path

2.4. Become familiar with the various options for the ls command. Give examples of listing directories using different keys. Explain the information displayed on the terminal using the -l and -a switches.

```
root@CsnKhai:/home/student# ls -a algorithms/
.  ..  fibonacci  fibonacci.c  fibonacci.cpp  fibonacci.py  hello.py
root@CsnKhai:/home/student# ls -l algorithms/
total 20
-rwxr-xr-x 1 root root 7326 Feb 16 21:29 fibonacci
-rw-r--r-- 1 root root 206 Feb 16 21:29 fibonacci.c
-rw-r--r-- 1 root root 85 Feb 16 21:27 fibonacci.cpp
-rw-r--r-- 1 root root 0 Feb 16 20:37 fibonacci.py
-rw-r--r-- 1 root root 37 Feb 16 20:22 hello.py
root@CsnKhai:/home/student# ls -l -author
total 48K
-rw-r--r-- 1 student 220 Sep 15 2015 .bash_logout
-rw-r--r-- 1 student 675 Feb 16 07:14 .profile
-rw-r--r-- 1 student 3.6K Feb 16 07:14 .bashrc
-rw----- 1 student 103 Feb 16 07:14 .bash_history
-rw-r--r-- 1 root 16 Feb 16 08:19 text
drwxr-xr-x 3 root 4.0K Feb 16 08:25 ..
drwxr-xr-x 2 root 4.0K Feb 16 08:31 bash-file
drwx----- 2 student 4.0K Feb 16 08:45 .cache
-rw-r--r-- 1 root 12 Feb 16 20:13 output
-rw-r--r-- 1 root 17 Feb 16 20:17 homework1
-rw-r--r-- 1 root 0 Feb 16 20:36 class1
drwxr-xr-x 5 student 4.0K Feb 16 20:36 .
drwxr-xr-x 2 root 4.0K Feb 16 21:30 algorithms
root@CsnKhai:/home/student#
```

2.5. Perform the following sequence of operations

- create a subdirectory in the home directory;
- in this subdirectory create a file containing information about directories located in the root directory (using I/O redirection operations);
- view the created file;

- copy the created file to your home directory using relative and absolute addressing

```
root@CsnKhai:/# ls -al > /home/test/root-listings
root@CsnKhai:/# cat /home/test/root-listings
total 80
drwxr-xr-x 21 root root 4096 Sep 15 2015 .
drwxr-xr-x 21 root root 4096 Sep 15 2015 ..
drwxr-xr-x 2 root root 4096 Sep 15 2015 bin
drwxr-xr-x 3 root root 4096 Sep 15 2015 boot
drwxr-xr-x 14 root root 4000 Feb 16 19:42 dev
drwxr-xr-x 83 root root 4096 Feb 16 21:16 etc
drwxr-xr-x 4 root root 4096 Feb 16 21:53 home
lrwxrwxrwx 1 root root 33 Sep 15 2015 initrd.img -> boot/initrd.img-3.13.0-63-generic
drwxr-xr-x 22 root root 4096 Feb 16 21:16 lib
drwx----- 2 root root 16384 Sep 15 2015 lost+found
drwxr-xr-x 2 root root 4096 Sep 15 2015 media
drwxr-xr-x 2 root root 4096 Apr 10 2014 mnt
drwxr-xr-x 2 root root 4096 Sep 15 2015 opt
dr-xr-xr-x 78 root root 0 Feb 16 19:42 proc
drwx----- 5 root root 4096 Sep 15 2015 root
drwxr-xr-x 16 root root 540 Feb 16 21:16 run
drwxr-xr-x 2 root root 4096 Sep 15 2015 sbin
drwxr-xr-x 2 root root 4096 Sep 15 2015 srv
dr-xr-xr-x 13 root root 0 Feb 16 19:42 sys
drwxrwxrwt 2 root root 4096 Feb 16 21:29 tmp
drwxr-xr-x 10 root root 4096 Sep 15 2015 usr
drwxr-xr-x 11 root root 4096 Sep 15 2015 var
lrwxrwxrwx 1 root root 30 Sep 15 2015 vmlinuz -> boot/vmlinuz-3.13.0-63-generic
root@CsnKhai:/# _
```

```
root@CsnKhai:/# cp /home/test/root-listings /home
root@CsnKhai:/# cd /home
root@CsnKhai:/home# cp test/root-listings /home
root@CsnKhai:/home#
```

- Delete the previously created subdirectory with the file requesting removal;
- Delete the file copied to the home directory.

```
root@CsnKhai:/home# rm -r test/
root@CsnKhai:/home# rm root-listings
root@CsnKhai:/home#
```

2.6. Perform the following sequence of operations:

- create a subdirectory test in the home directory;
- copy the .bash_history file to this directory while changing its name to labwork2;

```

root@CsnKhai:/home# cp student/.bash_history test;mv test/.bash_history test/lab
work2
root@CsnKhai:/home# ls -al test
total 12
drwxr-xr-x 2 root root 4096 Feb 16 23:31 .
drwxr-xr-x 4 root root 4096 Feb 16 23:28 ..
-rw----- 1 root root 103 Feb 16 23:31 labwork2
root@CsnKhai:/home# cat test/labwork2
sudo su
top
sudo update.rc ssh defaults
sudo update-rc.d ssh defaults
sudo reboot
sudo shutdown -h now
root@CsnKhai:/home#

```

- create a hard and soft link to the labwork2 file in the test subdirectory

```

root@CsnKhai:/home/test# ls
hardlink_labwork2 labwork2 softlink_labwork2
root@CsnKhai:/home/test# cat labwork2
sudo su
top
sudo update.rc ssh defaults
sudo update-rc.d ssh defaults
sudo reboot
sudo shutdown -h now
ls -al
root@CsnKhai:/home/test#

```

- how to define soft and hard link, what do these concepts;

Hard Link:

A hard link acts as a copy (mirrored) of the selected file. It accesses the data available in the original file.

If the earlier selected file is deleted, the hard link to the file will still contain the data of that file.

Soft Link:

A soft link (also known as symbolic link) acts as a pointer or a reference to the file name. It does not access the data available in the original file.

If the earlier file is deleted, the soft link will be pointing to a file that does not exist anymore.

- rename the hard link file to hard_lnk_labwork2;
- rename the soft link file to symb_lnk_labwork2 file;

```

root@CsnKhai:/home/test# mv hardlink_labwork2 hard_link_labwork2
root@CsnKhai:/home/test# mv softlink_labwork2 symb_link_labwork2
root@CsnKhai:/home/test# ls
hard_link_labwork2 labwork2 symb_link_labwork2
root@CsnKhai:/home/test# _

```

- then delete the labwork2. What changes have occurred and why?

```

root@CsnKhai:/home/test# rm labwork2
root@CsnKhai:/home/test# cat symb_link_labwork2
cat: symb_link_labwork2: No such file or directory
root@CsnKhai:/home/test# cat hard_link_labwork2
sudo su
top
sudo update-rc ssh defaults
sudo update-rc.d ssh defaults
sudo reboot
sudo shutdown -h now
ls -al
root@CsnKhai:/home/test# _

```

After deleting labwork2 the symbolic link is no more existent because symbolic is a pointer to a file and when the latter is deleted the symbolic link cannot point to a file that does not exist.

2.7. Using the locate utility; find all files that contain the squid and traceroute sequence.

```

student@CsnKhai:~$ locate squid
student@CsnKhai:~$ locate traceroute
/etc/alternatives/traceroute6
/etc/alternatives/traceroute6.8.gz
/lib/modules/3.13.0-63-generic/kernel/drivers/tty/n_tracerouter.ko
/usr/bin/traceroute6
/usr/bin/traceroute6.iputils
/usr/share/man/man8/traceroute6.8.gz
/usr/share/man/man8/traceroute6.iputils.8.gz
/var/lib/dpkg/alternatives/traceroute6
student@CsnKhai:~$ _

```

2.8. Determine which partitions are mounted in the system, as well as the types of these partitions

```

root@CsnKhai:/home# mount
/dev/sda1 on / type ext4 (rw,errors=remount-ro)
proc on /proc type proc (rw,noexec,nosuid,nodev)
sysfs on /sys type sysfs (rw,noexec,nosuid,nodev)
none on /sys/fs/cgroup type tmpfs (rw)
none on /sys/fs/fuse/connections type fusectl (rw)
none on /sys/kernel/debug type debugfs (rw)
none on /sys/kernel/security type securityfs (rw)
udev on /dev type devtmpfs (rw,mode=0755)
devpts on /dev/pts type devpts (rw,noexec,nosuid,gid=5,mode=0620)
tmpfs on /run type tmpfs (rw,noexec,nosuid,size=10%,mode=0755)
none on /run/lock type tmpfs (rw,noexec,nosuid,nodev,size=5242880)
none on /run/shm type tmpfs (rw,nosuid,nodev)
none on /run/user type tmpfs (rw,noexec,nosuid,nodev,size=104857600,mode=0755)
none on /sys/fs/pstore type pstore (rw)
systemd on /sys/fs/cgroup/systemd type cgroup (rw,noexec,nosuid,nodev,none,name=systemd)

```

```

root@CsnKhai:/home/student# df -aTh

```

Filesystem	Type	Size	Used	Avail	Use%	Mounted on
/dev/sda1	ext4	1.5G	1.1G	282M	80%	/
proc	proc	0	0	0	-	/proc
sysfs	sysfs	0	0	0	-	/sys
none	tmpfs	4.0K	0	4.0K	0%	/sys/fs/cgroup
none	fusectl	0	0	0	-	/sys/fs/fuse/connections
none	debugfs	0	0	0	-	/sys/kernel/debug
none	securityfs	0	0	0	-	/sys/kernel/security
udev	devtmpfs	112M	4.0K	112M	1%	/dev
devpts	devpts	0	0	0	-	/dev/pts
tmpfs	tmpfs	25M	384K	24M	2%	/run
none	tmpfs	5.0M	0	5.0M	0%	/run/lock
none	tmpfs	121M	0	121M	0%	/run/shm
none	tmpfs	100M	0	100M	0%	/run/user
none	pstore	0	0	0	-	/sys/fs/pstore
systemd	cgroup	0	0	0	-	/sys/fs/cgroup/systemd

```

root@CsnKhai:/home/student# _

```

```

root@CsnKhai:/dev# lsblk -f
NAME FSTYPE LABEL MOUNTPOINT
sda
└─sda1 ext4 /
sr0
root@CsnKhai:/dev#

```

2.9. Count the number of lines containing a given sequence of characters in a given file

```

root@CsnKhai:/home/student# grep Linux text | wc -l
6
root@CsnKhai:/home/student#

```

2.10. Using the find command, find all files in the /etc directory containing the host character sequence.

```
root@CsnKhai:/# find /etc -name host*
/etc/hosts
/etc/hosts.allow
/etc/init/hostname.conf
/etc/hostname
/etc/hosts.deny
/etc/host.conf
root@CsnKhai:/# _
```

2.11. List all objects in /etc that contain the ss character sequence. How can I duplicate a similar command using a bunch of grep

```
/etc/nsswitch.conf
/etc/insserv.conf.d
/etc/dbus-1/session.conf
/etc/dbus-1/session.d
/etc/passwd
/etc/X11/Xsession.d
/etc/X11/Xsession.d/99upstart
/etc/X11/Xsession.d/00upstart
/etc/sysctl.d/10-console-messages.conf
/etc/libnl-3/classid
/etc/bash_completion.d/insserv
/etc/insserv.conf
/etc/rc1.d/K20ssh
/etc/rc4.d/S20ssh
/etc/rc6.d/K20ssh
student@CsnKhai:~$ find /etc | grep ss
```

Command executed is: find /etc | grep ss

2.12 Organize a screen-by-screen print of the contents of the /etc directory. Hint: You must use stream redirection operations

```

-rw----- 1 root root      809 Feb 17 13:33 shadow-
-rw-r--r-- 1 root root       73 Sep 15 2015 shells
drwxr-xr-x 2 root root    4096 Sep 15 2015 skel
drwxr-xr-x 2 root root    4096 Sep 15 2015 ssh
drwxr-xr-x 4 root root    4096 Sep 15 2015 ssl
-rw-r--r-- 1 root root      21 Sep 15 2015 subgid
-rw----- 1 root root       0 Sep 15 2015 subgid-
-rw-r--r-- 1 root root      21 Sep 15 2015 subuid
-rw----- 1 root root       0 Sep 15 2015 subuid-
-r--r----- 1 root root     745 Feb 10 2014 sudoers
drwxr-xr-x 2 root root    4096 Sep 15 2015 sudoers.d
-rw-r--r-- 1 root root   2084 Apr  1 2013 sysctl.conf
drwxr-xr-x 2 root root    4096 Sep 15 2015 sysctl.d
drwxr-xr-x 3 root root    4096 Sep 15 2015 systemd
drwxr-xr-x 2 root root    4096 Sep 15 2015 terminfo
-rw-r--r-- 1 root root       8 Sep 15 2015 timezone
-rw-r--r-- 1 root root   1260 Jul  1 2013 ucf.conf
drwxr-xr-x 4 root root    4096 Sep 15 2015 udev
drwxr-xr-x 3 root root    4096 Sep 15 2015 ufw
-rw-r--r-- 1 root root     321 Jun 20 2013 updatedb.conf
drwxr-xr-x 3 root root    4096 Sep 15 2015 update-manager
drwxr-xr-x 2 root root    4096 Sep 15 2015 update-motd.d
-rw-r--r-- 1 root root     222 Apr 11 2014 upstart-xsessions
drwxr-xr-x 2 root root    4096 Sep 15 2015 vim
lrwxrwxrwx 1 root root      23 Sep 15 2015 vtrgb -> /etc/alternatives/vtrgb
-rw-r--r-- 1 root root   4812 Oct 30 2014 wgetrc
drwxr-xr-x 4 root root    4096 Sep 15 2015 X11
drwxr-xr-x 2 root root    4096 Sep 15 2015 xml
-rw-r--r-- 1 root root     349 Jun 26 2012 zsh_command_not_found
root@CsnKhai:/home# ls -l /etc > /dev/console

```

Command executed was: `ls -l /ect > /dev/console`

2.13 What are the types of devices and how to determine the type of device? Give examples.

There are two types of devices:

- character devices: talks to devices in a character by character (1 byte at a time);
- block devices: talk to devices 1 block at a time (1 block = 512 bytes to 32KB);

```

root@CsnKhai:/dev# head /home/student/device-listing
total 0
crw----- 1 root    root      10, 235 Feb 19 03:41 autofs
drwxr-xr-x 2 root    root      580 Feb 19 03:41 block
drwxr-xr-x 2 root    root       80 Feb 19 03:41 bsg
crw----- 1 root    root     10, 234 Feb 19 03:41 btrfs-control
drwxr-xr-x 3 root    root       60 Feb 19 03:41 bus
lrwxrwxrwx 1 root    root        3 Feb 19 03:41 cdrom -> sr0
drwxr-xr-x 2 root    root    3340 Feb 19 03:41 char
crw----- 1 root    root       5,  1 Feb 19 03:41 console
lrwxrwxrwx 1 root    root       11 Feb 19 03:41 core -> /proc/kcore
root@CsnKhai:/dev# ls -l > /home/student/device-listing

```

Command executed was: `ls -l > /home/student/device-listing`

We went to /dev directory then display the content of the directory while redetecting it to a file call device-listing `ls -l > /home/student/` then display the first line of that file using head command.

Note: we create that file at /home/student directory because it is a bad practice adding files in root directories.

2.15. List the first 5 directory files that were recently accessed in the /etc directory

```
root@CsnKhai:/home# ls -t /etc | head -5
mtab
shadow
passwd
passwd-
shadow-
root@CsnKhai:/home# _
```

Conclusion

In this paper, I report the result of my laboratory work using Linux operating system (Ubuntu distribution) on a virtual machine (virtualBox). The goal was to acquire knowledge on Linux commands to look up user information while completing this laboratory work I was able to change user information such as full name, home phone number and password.

The labwork consists of two parts; the first is to lookup user information using command like finger, and change those information-using commands like chfn or passwd. The second part consist of mastering the art of traveling through

user directory, checking file type, examine hard and symbolic links, checking for devices and their type, checking for file system etc.

After completion, it is with a brain full of important Linux tools to navigate user directories, change user information and track changing into Linux OS. With all being said, I can proudly say that the goal and objectives set in the beginning of this work is successfully achieved.