

RHEL 124
Lab Journal
CIT 217
Chaz Davis

BCTC
Spring 2020

February 18, 2020

Contents

1 RHEL 124 Labs 1 & 2	2
2 RHEL 124 Labs 3 & 4	7
3 RHEL 124 Labs 5 & 6	10
4 RHEL 124 Labs 7, 8, & 9	13

Lab 1

RHEL Labs 1 & 2

RHEL 124 Labs 1 & 2
Spring 2020

CIT 217
Chaz Davis

Part 1: Chapter 1 Questions

- i) Provide the output of a sucessful password change from the CLI

```
student@desktop1:~  
File Edit View Search Terminal Help  
[student@desktop1 ~]$ passwd  
Changing password for user student.  
Changing password for student.  
(current) UNIX password:  
New password:  
Retype new password:  
passwd: all authentication tokens updated successfully.  
[student@desktop1 ~]$ █
```

- ii) From the CLI, enter the command whoami and provide the output

```
student@desktop1:~  
File Edit View Search Terminal Help  
[student@desktop1 ~]$ whoami  
student  
[student@desktop1 ~]$ █
```

- iii) From the CLI, enter the command date +%D-%r and provide the output

```
student@desktop1:~  
File Edit View Search Terminal Help  
[student@desktop1 ~]$ date +%D-%r  
02/01/20-05:34:02 PM  
[student@desktop1 ~]$
```

- iv) Using File, determine what type of file is /etc./networks

```
student@desktop1:~  
File Edit View Search Terminal Help  
[student@desktop1 ~]$ file /etc/networks  
/etc/networks: ASCII text  
[student@desktop1 ~]$ █
```

v) Provide the top 5 lines of the file /etc/resolv.conf

```
student@desktop1:~$ head -n 5 /etc/resolv.conf
File Edit View Search Terminal Help
[student@desktop1 ~]$ head -n 5 /etc/resolv.conf
# Generated by NetworkManager
domain example.com
search example.com
nameserver 172.25.254.254
[student@desktop1 ~]$ █
```

vi) List your CLI history. Provide the output

```
File Edit View Search Terminal Help
[student@desktop1 ~]$ history
1 rm ../../.bash_history
2 exit
3 ip addr
4 ip addr
5 ncclient -r
6 sudo su
7 ip addr
8 ip addr
9 ip addr
10 ip addr
11 sudo su
12 ls
13 cd ..
14 ls
15 cd Do
16 cd Documents/
17 ls
18 cd ..
19 ls
20 cd Templates/
21 ls
22 cd ..
23 ls
24 df -h
25 fdisk
26 fdisk /dev/sdb
27 sudo su
28 touch tv_season1_episode1.ogg
29 touch tv_season1_episode2.ogg
30 touch tv_season1_episode3.ogg
31 touch tv_season1_episode4.ogg
32 touch tv_season1_episode5.ogg
33 touch tv_season1_episode6.ogg
34 touch tv_season2_episode6.ogg
```

(a) CLI History pg 1

```
File Edit View Search Terminal Help
33 touch tv_season1_episode6.ogg
34 touch tv_season2_episode6.ogg
35 touch tv_season2_episode5.ogg
36 touch tv_season2_episode4.ogg
37 touch tv_season2_episode3.ogg
38 touch tv_season2_episode2.ogg
39 touch tv_season2_episode1.ogg
40 clear
41 ls
42 touch mystery_chapter1
43 touch mystery_chapter1.odf
44 touch mystery_chapter3.odf
45 touch mystery_chapter4.odf
46 touch mystery_chapter6.odf
47 touch mystery_chapter7.odf
48 touch mystery_chapter8.odf
49 clear
50 ls
51 rm mystery_chapter1
52 clear
53 ls -l
54 cd vide
55 cd Videos/
56 clear
57 mkdir Season1/
58 passwd
59 whoami
60 date %D-%r
61 file /etc/networks
62 head -n 5 /etc/resolv.conf
63 cat /etc/resolv.conf
64 head -n 5 /etc/resolv.conf
65 exit
66 history
[student@desktop1 ~]$ █
```

(b) CLI History pg 2

Figure 1.1: Listing CLI History

Part 2: Chapter 2 Questions

- i) List the contents of /etc/sysctl.d/ using the long listing format. Provide the output

```
[student@desktop1 ~]$ ls -l /etc/sysctl.d/
total 0
-rwxrwxrwx. 1 root root 14 May  6 2014 99-sysctl.conf -> ../sysctl.conf
[student@desktop1 ~]$ ls -al /etc/sysctl.d/
total 12
drwxr-xr-x.  2 root root  27 May  6 2014 .
drwxr-xr-x. 133 root root 8192 Feb  1 17:25 ..
drwxrwxrwx.  1 root root  14 May  6 2014 99-sysctl.conf -> ../sysctl.conf
[student@desktop1 ~]$ █
```

- ii) Create Two file, your firstname and your lastname, then move them to the Documents directory. List the contents of the Documents directory. Provide the output.

```
[student@desktop1 ~]$ touch Chaz Davis
[student@desktop1 ~]$ mv Chaz Documents/ && mv Davis Documents/
[student@desktop1 ~]$ ls -l Documents/
total 0
-rw-rw-r--. 1 student student 0 Feb  1 17:47 Chaz
-rw-rw-r--. 1 student student 0 Feb  1 17:47 Davis
[student@desktop1 ~]$
```

- iii) From your home directory, create a directory named CIT217 then list the contents of your home directory. Provide the Output.

```
[student@desktop1 ~]$ mkdir CIT217
[student@desktop1 ~]$ ls -l
total 0
drwxrwxr-x. 2 student student  6 Feb  1 18:00 CIT217
drwxr-xr-x. 2 student student  6 Apr 20 2015 Desktop
drwxr-xr-x. 2 student student 29 Feb  1 17:54 Documents
drwxr-xr-x. 2 student student  6 Apr 20 2015 Downloads
drwxr-xr-x. 2 student student  6 Apr 20 2015 Music
-rw-rw-r--. 1 student student 0 Jan 25 2019 mystery_chapter1.odf
-rw-rw-r--. 1 student student 0 Jan 25 2019 mystery_chapter2.odf
-rw-rw-r--. 1 student student 0 Jan 25 2019 mystery_chapter3.odf
-rw-rw-r--. 1 student student 0 Jan 25 2019 mystery_chapter4.odf
-rw-rw-r--. 1 student student 0 Jan 25 2019 mystery_chapter5.odf
-rw-rw-r--. 1 student student 0 Jan 25 2019 mystery_chapter6.odf
-rw-rw-r--. 1 student student 0 Jan 25 2019 mystery_chapter7.odf
-rw-rw-r--. 1 student student 0 Jan 25 2019 mystery_chapter8.odf
drwxr-xr-x. 2 student student  6 Apr 20 2015 Pictures
drwxr-xr-x. 2 student student  6 Apr 20 2015 Public
drwxr-xr-x. 2 student student  6 Apr 20 2015 Templates
-rw-rw-r--. 1 student student 0 Jan 25 2019 tv_season1_episode1.ogg
-rw-rw-r--. 1 student student 0 Jan 25 2019 tv_season1_episode2.ogg
-rw-rw-r--. 1 student student 0 Jan 25 2019 tv_season1_episode3.ogg
-rw-rw-r--. 1 student student 0 Jan 25 2019 tv_season1_episode4.ogg
-rw-rw-r--. 1 student student 0 Jan 25 2019 tv_season1_episode5.ogg
-rw-rw-r--. 1 student student 0 Jan 25 2019 tv_season1_episode6.ogg
-rw-rw-r--. 1 student student 0 Jan 25 2019 tv_season2_episode1.ogg
-rw-rw-r--. 1 student student 0 Jan 25 2019 tv_season2_episode2.ogg
-rw-rw-r--. 1 student student 0 Jan 25 2019 tv_season2_episode3.ogg
-rw-rw-r--. 1 student student 0 Jan 25 2019 tv_season2_episode4.ogg
-rw-rw-r--. 1 student student 0 Jan 25 2019 tv_season2_episode5.ogg
-rw-rw-r--. 1 student student 0 Jan 25 2019 tv_season2_episode6.ogg
drwxr-xr-x. 3 student student 20 Jan 25 2019 Videos
[student@desktop1 ~]$
```

- iv) Run the command touch test1..4in the CLI. Using a wildcard, copy all the test files to the Documents directory. List the contents of the Documents directory and provide the output

```
File Edit View Search Terminal Help
[student@desktop1 ~]$ touch test{1..4}
[student@desktop1 ~]$ cp test* Documents/
[student@desktop1 ~]$ cd Documents/
[student@desktop1 Documents]$ ls -l
total 0
-rw-rw-r--. 1 student student 0 Feb 1 17:53 Chaz
-rw-rw-r--. 1 student student 0 Feb 1 17:53 Davis
-rw-rw-r--. 1 student student 0 Feb 1 18:05 test1
-rw-rw-r--. 1 student student 0 Feb 1 18:05 test2
-rw-rw-r--. 1 student student 0 Feb 1 18:05 test3
-rw-rw-r--. 1 student student 0 Feb 1 18:05 test4
[student@desktop1 Documents]$
```

Lab 2

Labs 3 & 4

RHEL 124 Labs 3 & 4
Spring 2020

CIT 217
Chaz Davis

Part 1: Chapter 3 Questions

i) Man Page of tar

```
NAME
tar - manual page for tar 1.26

SYNOPSIS
tar [OPTION...] [FILE]...

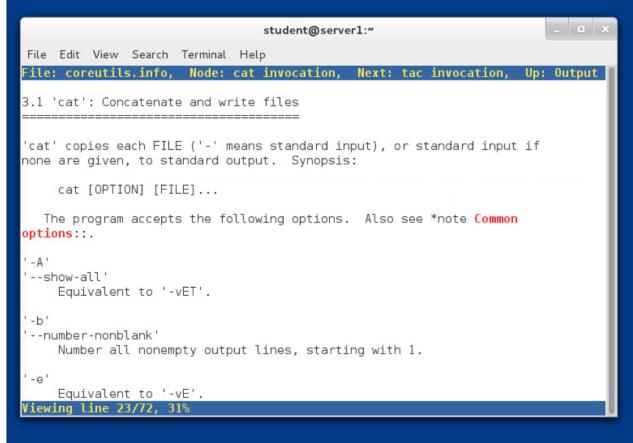
DESCRIPTION
GNU 'tar' saves many files together into a single tape or disk archive,
and can restore individual files from the archive.

Note that this manual page contains just very brief description (or
more like a list of possible functionality) originally generated by the
help2man utility. The full documentation for tar is maintained as a
Texinfo manual. If the info and tar programs are properly installed at
your site, the command 'info tar' should give you access to the com-
plete manual.

EXAMPLES
tar -cf archive.tar foo bar
# Create archive.tar from files foo and bar.

tar -tvf archive.tar
Manual page tar(1) line 2 (press h for help or q to quit)
```

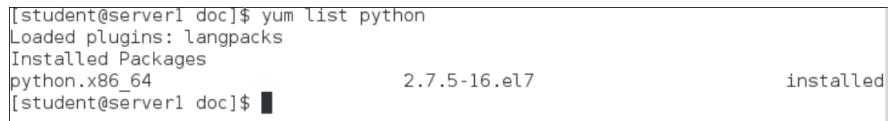
ii) Pinfo of cat



A screenshot of a terminal window titled "student@server1:~". The window displays the man page for the "cat" command. The text is as follows:

```
student@server1:~  
File Edit View Search Terminal Help  
File: coreutils.info, Node: cat invocation, Next: tac invocation, Up: Output  
3.1 'cat': Concatenate and write files  
=====  
'cat' copies each FILE ('-' means standard input), or standard input if  
none are given, to standard output. Synopsis:  
cat [OPTION] [FILE]...  
The program accepts the following options. Also see *note Common  
options...:  
'-A'  
'--show-all'  
      Equivalent to '-vET'.  
'-b'  
'--number-nonblank'  
      Number all nonempty output lines, starting with 1.  
'-e'  
      Equivalent to '-vE'.  
Viewing line 23/72, 31%
```

iii) Listing Yum Pkgs with python



```
[student@server1 doc]$ yum list python  
Loaded plugins: langpacks  
Installed Packages  
python.x86_64          2.7.5-16.el7      installed
```

Part 2: Chapter 4 Questions

i) Edit lab.txt with vim

```
[student@server1 ~]$ head -n 10 lab.txt
Chaz Davis
total 40
drwx-----, 17 student student 4096 Jan 30 15:38 .
drwxr-xr-x, 3 root root 20 Jan 6 2015 ..
-rw-----, 1 student student 284 Jan 30 15:37 .bash_history
-rw-r--r--, 1 student student 18 Jan 29 2014 .bash_logout
-rw-r--r--, 1 student student 193 Jan 29 2014 .bash_profile
-rw-r--r--, 1 student student 231 Jan 29 2014 .bashrc
drwx-----, 11 student student 4096 Jan 30 15:36 .cache
drwxr-xr-x, 14 student student 4096 Apr 20 2015 .config
[student@server1 ~]$ █
```

ii) delete 5th line save and output 10 lines

```
[student@server1 ~]$ head -n 10 lab.txt
Chaz Davis
total 40
drwx-----, 17 student student 4096 Jan 30 15:38 .
drwxr-xr-x, 3 root root 20 Jan 6 2015 ..
-rw-r--r--, 1 student student 18 Jan 29 2014 .bash_logout
-rw-r--r--, 1 student student 193 Jan 29 2014 .bash_profile
-rw-r--r--, 1 student student 231 Jan 29 2014 .bashrc
drwx-----, 11 student student 4096 Jan 30 15:36 .cache
drwxr-xr-x, 14 student student 4096 Apr 20 2015 .config
drwxr-xr-x, 2 student student 6 Apr 20 2015 Desktop
[student@server1 ~]$ █
```

iii) Visual Block delete column of months

```
[student@server1 ~]$ vim lab.txt
[student@server1 ~]$ head -n 10 lab.txt
Chaz Davis
total 40
drwx-----, 17 student student 4096 30 15:38 .
drwxr-xr-x, 3 root root 20 6 2015 ..
-rw-r--r--, 1 student student 18 29 2014 .bash_logout
-rw-r--r--, 1 student student 193 29 2014 .bash_profile
-rw-r--r--, 1 student student 231 29 2014 .bashrc
drwx-----, 11 student student 4096 30 15:36 .cache
drwxr-xr-x, 14 student student 4096 20 2015 .config
drwxr-xr-x, 2 student student 6 20 2015 Desktop
[student@server1 ~]$ █
```

Lab 3

RHEL 124 Labs 5 & 6

RHEL124 Labs 5 & 6
Spring 2020

CIT 217
Chaz Davis

Chapter 5 Questions

- i) Run the `firewall-cmd --list-all` with the necessary privilege escalation. Output Provided on Pg. 11 See Fig 3.1a
- ii) Switch user to root. Run the command `whoami`. Output Provided on Pg. 11 See Fig 3.1b
- iii) Add your BCTC credential as a user. Change the user's password. Provide the output of the encrypted file including your user credential. Output Provided on Pg. 11 See Fig 3.1c
- iv) Change the password policy for your user credential to require a new password every 180 days. Output Provided on Pg. 11 See Fig 3.1d
- v) Add a group named students with the GID of 50000. Add your user to the students group. Provide the output of the group file including the group and your user's membership. Output Provided on Pg. 11 See Fig 3.1e

```
[student@server1 ~]$ sudo firewall-cmd --list-all
[sudo] password for student:
public (default, active)
interfaces: eth0
sources:
services: dhcpcv6-client ssh
ports:
masquerade: no
forward-ports:
icmp-blocks:
rich rules:
[student@server1 ~]$
```

(a) Output after running
firewall-cmd --list-all

```
[student@server1 ~]$ su -
Password:
Last login: Mon Dec 21 10:01:50 EST 2015 on pts/0
[root@server1 ~]# whoami
root
[root@server1 ~]#
```

(b) Output after running whoami as root

```
root@server1 ~]# useradd cdavis0532
root@server1 ~]# passwd cdavis0532
Changing password for user cdavis0532.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a dictionary word
Retype new password:
passwd: all authentication tokens updated successfully.
root@server1 ~]# tail -5 /etc/passwd
pulse:x:171:171:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:993:991::/run/gnome-initial-setup:/sbin/nologin
cpdump:x:72:72::/sbin/nologin
cdavis0532:x:1001:1001::/home/cdavis0532:/bin/bash
[root@server1 ~]#
```

(c) encrypted password for my user

```
[root@server1 ~]# date -d "+180"
date: invalid date '+180'
[root@server1 ~]# date -d "+180 days"
Tue Aug 4 09:04:04 EDT 2020
[root@server1 ~]# chage -E 2020-08-04 cdavis0532
[root@server1 ~]#
```

(d) 180 day password policy

```
root@server1 ~]# chage -E 2020-08-04 cdavis0532
root@server1 ~]# groupadd -g 50000 students
root@server1 ~]# usermod -G students cdavis0532
root@server1 ~]# tail -5 /etc/group
stapdev:x:158:
cpdump:x:72:
locate:x:21:
davis0532:x:1001:
students:x:50000:cdavis0532
root@server1 ~]#
```

(e) Membership of the students group. with a 50000 GID

Figure 3.1: The Screenshots from Chapter 5

Chapter 6 Questions

- i) Switch user to root. Create a file named for your last name. See Fig 3.2a on Pg. 12. Change the permissions of the file to forbid other users from accessing it. Provide the detailed list output of the directory. Output Provided on Pg. 12 See Fig 3.2b
- ii) Add your BCTC credential as a user. Switch user to your credential. Create a file named test1. Change your umask to 027. Create a file named test2. Provide the detailed list output of the directory. Output Provided on Pg. 12 See Fig 3.2c

```
[root@server1 ~]# touch davis
[root@server1 ~]# l
bash: l: command not found...
[root@server1 ~]# ls
anaconda-ks.cfg  davis
[root@server1 ~]# ls -al
total 48
dr-xr-x---. 6 root root 4096 Feb  6 08:07 .
drwxr-xr-x. 17 root root 4096 Feb  6 07:41 ..
-rw-----. 1 root root 8619 May  6 2014 anaconda-ks.cfg
-rw-----. 1 root root 212 Dec 21 2015 .bash_history
-rw-r--r--. 1 root root 18 Dec 28 2013 .bash_logout
-rw-r--r--. 1 root root 176 Dec 28 2013 .bash_profile
-rw-r--r--. 1 root root 176 Dec 28 2013 .bashrc
drwxr-xr-x. 4 root root 29 Aug 24 2015 .cache
drwxr-xr-x. 4 root root 79 Aug 24 2015 .config
-rw-r--r--. 1 root root 100 Dec 28 2013 .cshrc
-rw-r--r--. 1 root root 0 Feb  6 08:07 davis
drwx-----. 3 root root 24 Aug 24 2015 .dbus
drwx-----. 2 root root 28 Jan  6 2015 .ssh
-rw-r--r--. 1 root root 129 Dec 28 2013 .tcshrc
-rw-r--r--. 1 root root 64 Feb  6 07:48 .xauth4Bgj9v
```

(a) Detailed list view after creating the file davis

```
[root@server1 ~]# chmod 700 davis
[root@server1 ~]# ls -al
total 48
dr-xr-x---. 6 root root 4096 Feb  6 08:07 .
drwxr-xr-x. 17 root root 4096 Feb  6 07:41 ..
-rw-----. 1 root root 8619 May  6 2014 anaconda-ks.cfg
-rw-----. 1 root root 212 Dec 21 2015 .bash_history
-rw-r--r--. 1 root root 18 Dec 28 2013 .bash_logout
-rw-r--r--. 1 root root 176 Dec 28 2013 .bash_profile
-rw-r--r--. 1 root root 176 Dec 28 2013 .bashrc
drwxr-xr-x. 4 root root 29 Aug 24 2015 .cache
drwxr-xr-x. 4 root root 79 Aug 24 2015 .config
-rw-r--r--. 1 root root 100 Dec 28 2013 .cshrc
-rwx-----. 1 root root 0 Feb  6 08:07 davis
drwx-----. 3 root root 24 Aug 24 2015 .dbus
drwx-----. 2 root root 28 Jan  6 2015 .ssh
-rw-r--r--. 1 root root 129 Dec 28 2013 .tcshrc
-rw-----. 1 root root 64 Feb  6 07:48 .xauth4Bgj9v
[root@server1 ~]#
```

(b) Detailed List after changing file permissions

```
[root@server1 ~]# su - cdavis0532
[cdavis0532@server1 ~]$ touch test1
[cdavis0532@server1 ~]$ ls
test1
[cdavis0532@server1 ~]$ umask 027
[cdavis0532@server1 ~]$ touch test2
[cdavis0532@server1 ~]$ ls -ld
drwx----- 5 cdavis0532 cdavis0532 4096 Feb  6 08:13 .
[cdavis0532@server1 ~]$ ls -al
total 16
drwx----- 5 cdavis0532 cdavis0532 4096 Feb  6 08:13 .
drwxr-xr-x. 4 root  root   37 Feb  6 07:59 ..
-rw-r--r--. 1 cdavis0532 cdavis0532 18 Jan 29 2014 .bash_logout
-rw-r--r--. 1 cdavis0532 cdavis0532 193 Jan 29 2014 .bash_profile
-rw-r--r--. 1 cdavis0532 cdavis0532 231 Jan 29 2014 .bashrc
drwxrwxr-x. 3 cdavis0532 cdavis0532 17 Feb  6 08:10 .cache
drwxr-xr-x. 3 cdavis0532 cdavis0532 67 Feb  6 08:10 .config
drwxr-xr-x. 4 cdavis0532 cdavis0532 37 Jan  6 2015 .mozilla
-rw-rw-r--. 1 cdavis0532 cdavis0532  0 Feb  6 08:10 test1
-rw-r-----. 1 cdavis0532 cdavis0532  0 Feb  6 08:13 test2
[cdavis0532@server1 ~]$
```

(c) test1 and test2 files with their umasks updated

Figure 3.2: The Screenshots from Chapter 6

Lab 4

RHEL 124 Labs 7, 8, & 9

RHEL 124 Labs
Spring 2020

CIT 288
Chaz Davis

Part 1: Chapter 7 Questions

- i) Run the command `sleep 1000` in the background. Using the `ps` command, provide the output displaying it's still running.

I ran `sleep 1000` into the terminal.

I then ran the command `ctrl + z` to suspend the program and then `bg` to put it into the background.
Finally, I ran `ps` to show the running processes on the system. The output you can see in Fig. 4.1a on Pg. 14.

- ii) Abruptly terminate the sleep process that you created. Use the `ps` command to provide the output it is no longer running.

First, I brought the sleep command back from the background to the foreground using `fg`.

Then I used the key sequence `ctrl + c` to kill the foregrounded process.

Finally, I ran `ps` to show the running processes on the system. The output you can see in Fig. 4.1b on Pg. 14.

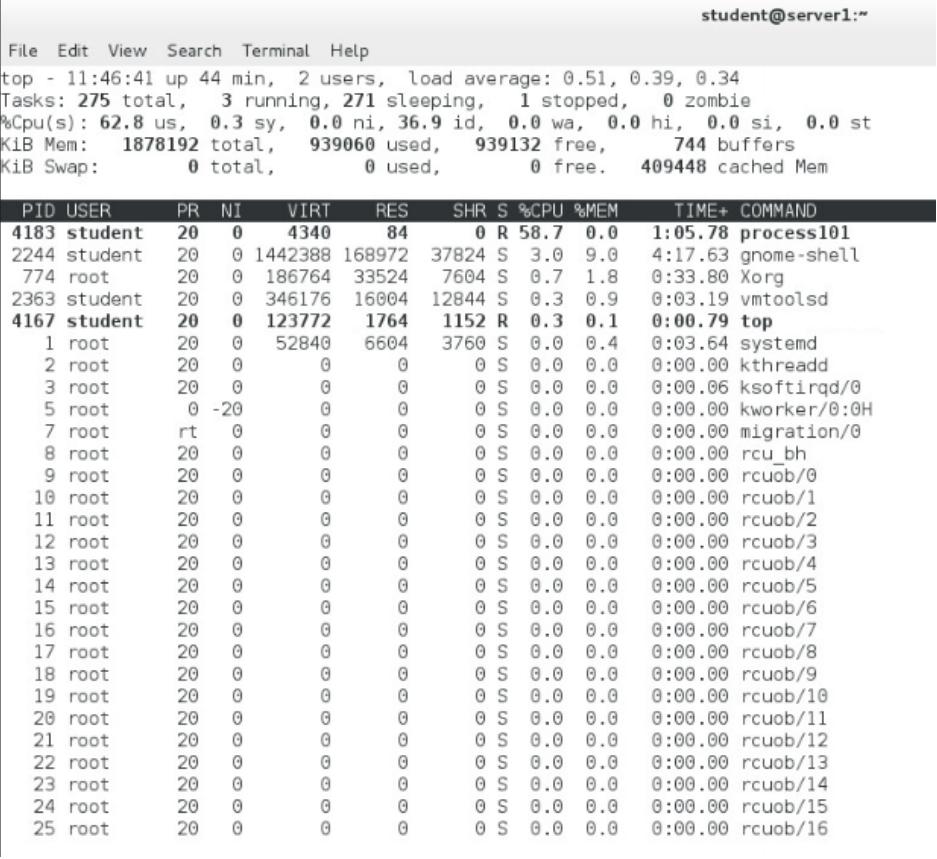
- iii) Provide the dynamic output of the top running processes on your system.

First I opened a terminal and then ran the command `top`, the output of which you can see in Fig. 4.1c on Pg. 14.

```
[student@server1 ~]$ sleep 1000
^Z
[2]+  Stopped                  sleep 1000
student@server1 ~]$ bg
[2]+ sleep 1000 &
student@server1 ~]$ ps
  PID TTY      TIME CMD
3360 pts/0    00:00:00 bash
3668 pts/0    00:00:00 tail
4207 pts/0    00:00:00 sleep
4222 pts/0    00:00:00 ps
```

```
[student@server1 ~]$ fg
sleep 1000
^C
[student@server1 ~]$ ps
  PID TTY      TIME CMD
3360 pts/0    00:00:00 bash
4244 pts/0    00:00:00 ps
[student@server1 ~]$ █
```

(a) Starting and verifying the Sleep 1000 process (b) Terminating the sleep process



The terminal window displays system load statistics and a detailed process list. The top section shows:

```
top - 11:46:41 up 44 min, 2 users, load average: 0.51, 0.39, 0.34
Tasks: 275 total, 3 running, 271 sleeping, 1 stopped, 0 zombie
%Cpu(s): 62.8 us, 0.3 sy, 0.0 ni, 36.9 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 1878192 total, 939060 used, 939132 free, 744 buffers
KiB Swap: 0 total, 0 used, 0 free. 409448 cached Mem
```

The bottom section is a table of processes:

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
4183	student	20	0	4340	84	0	R	58.7	0.0	1:05.78	process101
2244	student	20	0	1442388	168972	37824	S	3.0	9.0	4:17.63	gnome-shell
774	root	20	0	186764	33524	7604	S	0.7	1.8	0:33.80	Xorg
2363	student	20	0	346176	16004	12844	S	0.3	0.9	0:03.19	vmtoolsd
4167	student	20	0	123772	1764	1152	R	0.3	0.1	0:00.79	top
1	root	20	0	52840	6604	3760	S	0.0	0.4	0:03.64	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.06	ksoftirqd/0
5	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/0:0H
7	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	migration/0
8	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_bh
9	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/0
10	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/1
11	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/2
12	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/3
13	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/4
14	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/5
15	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/6
16	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/7
17	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/8
18	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/9
19	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/10
20	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/11
21	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/12
22	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/13
23	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/14
24	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/15
25	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/16

(c) Output of Top

Figure 4.1: Chapter 7 Screenshots

Part 2: Chapter 8 Questions

i) Provide the output of the system status for the service firewalld.

I went to the terminal on server1 and entered `sudo systemctl status firewalld`.

I then entered my password.

Finally, I was given the output of the status of the firewall daemon see Fig. 4.2a

ii) Is the service nfs enabled or disabled? Provide the output of its state.

After logging into the terminal and entering the command `sudo systemctl status nfs` and entering my credentials, we can now see in Fig. 4.2b that nfs is loaded but not active.

Alternatively, I could have run the command `sudo systemctl is-enabled nfs` that output is provided in Fig. 4.2c.

```
student@server1 ~]$ sudo systemctl status firewalld
firewalld.service - firewalld - dynamic firewall daemon
  Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled)
  Active: active (running) since Thu 2020-02-13 11:02:05 EST; 1h 6min ago
    Main PID: 642 (firewalld)
   CGroup: /system.slice/firewalld.service
           └─642 /usr/bin/python -Es /usr/sbin/firewalld --nofork --nopid

Feb 13 11:02:05 localhost systemd[1]: Started firewalld - dynamic firewall daemon.
student@server1 ~]$ █
```

(a) Firewalld status

```
student@server1 ~]$ sudo systemctl status nfs
[sudo] password for student:
nfs-server.service - NFS Server
  Loaded: loaded (/usr/lib/systemd/system/nfs-server.service; disabled)
  Active: inactive (dead)

[student@server1 ~]$ █
```

(b) nfs Status

```
[student@server1 ~]$ sudo systemctl is-enabled nfs
[sudo] password for student:
disabled
[student@server1 ~]$
```

(c) nfs is-enabled

Figure 4.2: Chapter 8 Screenshots

Part 3: Chapter 9 Questions

i) ssh to server1 then run the hostname command. Provide the output.

I logged into Desktop1 and opened a terminal. I then entered `ssh student@server1` after confirmation and key creation I was able to enter my password for the server account.

I then ran the command `hostname` the output of which you can see in Fig. 4.3a on Pg. 17.

ii) Edit the sshd config file. Disable root logins. Disable strict modes. Provide the output of the file where this was accomplished.

I logged into the server and used the command `sudo vim /etc/ssh/sshd_config`

I then, went down to the authentication section, and changed the yes to a no for both PermitRootLogin see Fig. 4.3b on Pg. 17 and for StrictModes. See Fig. 4.3c on Pg. 17.

Then to verify it took effect I used the command `less /etc/ssh/sshd_config` See Fig. 4.3d on Pg. 17.

iii) Generate an ssh key saved as your first name. Provide the output.

I used the command `ssh-keygen` and when prompted for file I told it to save as Chaz.

To verify this I used the command `cat chaz`, the output is displayed in Fig. 4.3e on Pg. 17.

```
[student@desktop1 ~]$ ssh student@server1
The authenticity of host 'server1 (172.25.1.11)' can't be established.
ECDSA key fingerprint is 65:4d:ac:8a:c9:58:82:b5:0c:91:c4:ef:a5:e6:f6:65.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
Warning: Permanently added 'server1,172.25.1.11' (ECDSA) to the list of known hosts.
student@server1's password:
Last login: Thu Feb 13 11:25:06 2020
[student@server1 ~]$ hostname
server1.example.com
[student@server1 ~]$
```

(a) Server1 hostname

Authentication:

```
#LoginGraceTime 2m
#PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

Authentication:

```
#LoginGraceTime 2m
#PermitRootLogin no
#StrictModes no
#MaxAuthTries 6
#MaxSessions 10
```

(b) PermitRootLogin

```
File Edit View Search Terminal Help
root@server1:~#
# Ciphers and keying
#RekeyLimit default none

# Logging
# obsoletes QuietMode and FascistLogging
#SyslogFacility AUTH
#syslogFacility AUTHPRIV
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin no
#StrictModes no
#MaxAuthTries 6
#MaxSessions 10

#RSAAuthentication yes
#PubkeyAuthentication yes

# The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
# but this is overridden so installations will only check .ssh/authorized_keys
#authorizedkeyFile .ssh/authorized_keys

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#RhostsRSAAuthentication no
# similar for protocol version 2
#HostbasedAuthentication no
```

(c) StrictModes

root@server1:~#

(d) Less sshd config

```
[root@server1 ~]# cat Chaz
-----BEGIN RSA PRIVATE KEY-----
MIIEpQIBAAKCAQEa2ESLc0LETE40qrdr8t0cRcP0t02drDBA9KxbGr05LDQt17Pl
21wbM10ula3fNLZ7e9IuFFnRsQ+h3vyW6dAmh0RWPrRhIKnjzbo6zaw9cTQrBgt
LtgEphdB/nLU2+a+QKM1i98MJmz2fP0hM58Jjig190kR+rh54CzsL+DUyZogUjE
C4KF0Ztpw0/yGSNjaPny/nJNx4R7Fwgn87uXP0NT73gTDk6XMcGivU/Pu/gbtjeJ
e6dJMyxsZa2tk0peMsVA7CLNUCAL6HlwvNrwb/sd6Mch008BPo+If02kr2C6ft
Chu3Nm2dkf0HuytgyZ6ML13tFI3jTcyBiMipwIDAQABaCIBAQCV9m0Lf2zuqgG5
lyo/3GPXsuC+EIRGBSyB9xNkIghl3eeovMM74YxSfvT08r014+GewjBontIS3Z
dcUftFpCbvS+Nc80xFhoo8L9xd3BzIHYTWita0ZQ059N/2fCYIpARVGFP0ZKc
72mMngdjT1IkDXg/1Q1FoTsSuUzzV1wV3Gteymal5v6Gq5/5Ha2w2rLoEMDw
JbSmd/cS33tg+0ssqaqjCnPn5zcQg3Y/4PzB2sXcAgQVj13D0mB0vJ8kL2KsuiXM
Ah+aHk61HnhE6swyT0b0GIVCh/e0Rp3EsPejgMt4a9lxyctrFgfJHr6av91w5
oZ/1M1dhAoG8APm/JPf8l0RUMU/CmdtTip2H6kwofc9g+9A00/H/tG2pbgiFBMOM
rCDNTv0AkseauNjj70owfJMgfI7zzqCY4Hvp6KBZCSMTQPI14zm8/8xt0YGe
yB8vLvxzhwYMDT/IbficafctsZ7AAWJgycCp89PNdv7DwGFQy@04jAaGBAN2u
5LYkJ6znYQpewwPxKR17lwRUwt1v89H7gvtwgRxx1lMyqoru23wbVv5Jvur
l6sK2QsvufE2yL3VTubn1azaaHcerPubAZX16zu93eC+/T/4surMwu2NTk0cqvjx
kJFywKGhoqzSD1HVVHI6cWz4TKU98680M1tALGmtAoGBAIQpVJjRHNIWTzDCTSb
ffzyBGvQnas1fmxSlo/NkjTkNu1d1cYcq0+LmUqReDTTzmrHBLdsxGB0okHmi
vAPq4UYPyND0awnDzW4CTnPu2n6BgkQ/txJ9re1k967a+VPSg9AUrGHbkd11B0a
B06hNkLITD8RJb0Tirxb30wTAoGP1cyw4ZH1vHn9jEjEHgOF9zsu4viwRlwHRS
kqL+xqvIzt8ZB0PMwqlD09B4udpph14-6qcEsBrWzTUL/SaCCwjssRos5owiy
R1jJbotdzwP/5hK9yb5f42u+YMMERu8mxULStgCnMwrcamxin7231QEElx0DWtG2
S96fbkCgYEaf3r0AJYi512lwENusoKneL6wF85nqyZTn4Pbq4riPMZyZw/EQEj
f3rQnjUumPqexve75yc8jnP+kz3nTpGRF/FmgLDmBcw5Uo0STCzeZzW6tR9G2
ay/20061c5oswTjKTDAbsz5/s0ef7/d93E14sD1LUUv7fwuXzj6c=
-----END RSA PRIVATE KEY-----
[root@server1 ~]
```

(e) RSA ouput of Chaz