Week 5
Lab Report
CIT 217
Chaz Davis

BCTC Spring 2020

February 13, 2020

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 ctr1 + 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. 1a on Pg. 2.

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.

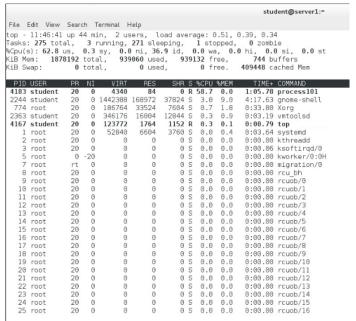
Finally, I ran $_{PS}$ to show the running processes on the system. The output you can see in Fig. 1b on Pg. 2.

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. 1c on Pg. 2.

```
student@server1 ~]$ fg
student@server1 ~]$ sleep 1000
                                                   sleep 1000
2]+ Stopped
                           sleep 1000
student@server1 ~]$ bg
2]+ sleep 1000 &
                                                   student@server1 ~]$ ps
student@server1 ~]$ ps
                                                    PID TTY
                                                                         TIME CMD
PID TTY
3360 pts/0
                 TIME CMD
                                                                     00:00:00 bash
                                                   3360 pts/0
             00:00:00 bash
                                                   4244 pts/0
                                                                    00:00:00 ps
3668 pts/0
             00:00:00 tail
                                                   student@server1 ~]$
4207 pts/0
             00:00:00 sleep
4222 pts/0
             00:00:00 ps
```

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



(c) Output of Top

Figure 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 systemct1 status firewalld. I then entered my password.

Finally, I was given the output of the status of the firewall daemon see Fig. 2a

ii) Is the service of 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. 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. 2c.

```
student@serverl ~]$ 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
eb 13 11:02:05 localhost systemd[1]: Started firewalld - dynamic firewall daemon.
student@server1 ~]$
                               (a) Firewalld status
 student@serveri ~]$ sudo systemcti status n†s
  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 ~]$
```

Figure 2: Chapter 8 Screenshots

(c) nfs is-enabled

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. 3a on Pg. 5.

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. 3b on Pg. 5 and for StrictModes. See Fig. 3c on Pg. 5.

Then to verify it took effect I used the command less /etc/ssh/sshd_config See Fig. 3d on Pg. 5.

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

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. 3e on Pg. 5.

```
[student@desktopl ~]$ ssh student@serverl
The authenticity of host 'serverl (172.25.1.11)' can't be established.
ECDSA key fingerprint is 65:4d:ac:8a:c9:58:82:b5:0c:9l: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 'serverl,172.25.1.11' (ECDSA) to the list of known hosts.
 student@serverl's password:
Last login: Thu Feb 13 11:25:06 2020
[student@serverl ~]$ hostname
  server1.example.com
[student@server1 ~]$
```

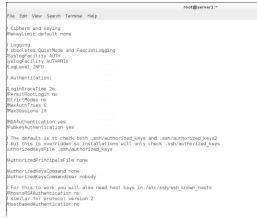
(a) Server1 hostname

```
Authentication:
#LoginGraceTime 2m
#<mark>Permit</mark>RootLogin n<mark>o</mark>
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

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

(b) PermitRootLogin

(c) StrictModes



(d) Less sshd config

(e) RSA ouput of Chaz

Figure 3: Chapter 9 screenshots