



---

# ASSIGNMENT 4-SCRIPTING

---

Written By: Daniel Piche



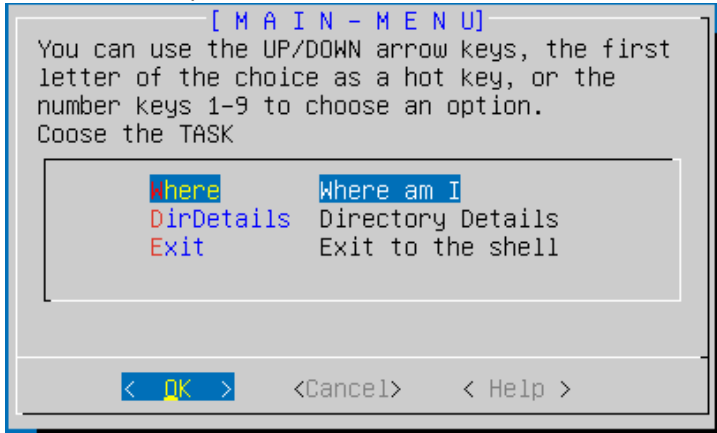
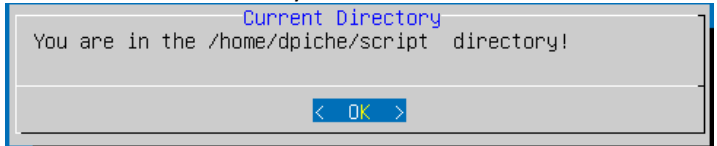
NOVEMBER 9, 2018  
INET3700-SERVER OS AND SCRIPTING  
NSCC-IT Campus

## Table of Contents

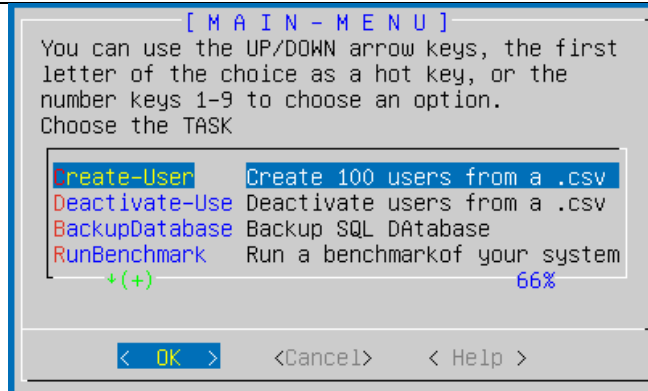
Introduction .....	2
Install Dialog.....	2
First Menu Box Script.....	2
INET3700 Menu Box Script .....	3
Benchmark Script with crontab .....	8
Crontab Backups .....	10
Snapshot .....	11
Gold Copy.....	11
Summary .....	11
Glossary.....	12
1 <sup>st</sup> Menu Box Script .....	12
2 <sup>nd</sup> Menu Box Script .....	13
Benchmark Script with crontab .....	16
Change Management Log .....	16
References .....	16
Helpful Links.....	17

## Introduction

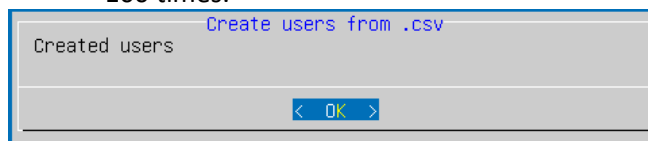
For the purposes of assignment 4 we will be working with shell scripts and crontab. Below is all the documentation for the assignment. This documentation will be used as a reference for the future, in the event that you are faced with the challenge of writing a script.

Activity List			
Project: Scripting			Date: 30/10/2018
Activity ID	Activity Name	Description of work	Responsibility
	Install Dialog	1) Before creating any menu box scripts you need to install "dialog" <code>sudo apt-get install dialog</code>	
Task 1	First Menu Box Script	<p>1) The screen shot below shows you the main menu to the script</p>  <p>2) If you select "where" you will get the path of the current directory.</p>  <p>3) If you select directory details you will get the menu below.</p> <p>4) The first line shows the owner of the current directory.</p> <p>5) The second line shows the permissions to the current directory.</p> <p>6) The third line shows the number of files in the current directory</p> <p>7) The fourth line shows the size and name of the biggest file.</p> <p>8) The last line shows the name of the biggest file.</p>	

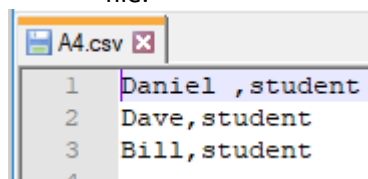
		<p style="text-align: center;"><a href="#">Directory Info</a></p> <pre> The owner of the current directory is :dpiche  Owner permissions are :# file: home/dpiche/script # owner: dpiche # group: dpiche user::rwx group::rwx other::r-x  The number of files in the current directory is :17  The name and size of the biggest file is :12K users.csv  The name of the biggest file is :users.csv </pre> <p style="text-align: center;">&lt; OK &gt;</p> <p>9) If you select "Exit" the menu will close and you will get a "Bye" message.</p> <pre> Bye dpiche@dbrmd213_01:~/script\$ dpiche@dbrmd213_01:~/script\$ _ </pre> <p>(CyberCiti, 2016, page 1)  (StackOverflow, 2018, page 1)  (StackExchange, 2018, page 1)  (Nix Craft, 2013, page 1)  <a href="#">See glossary for script</a></p>	
	INET3700 Menu Box Script	<p>1) Before being able to add users with a .csv file you first need to install a package.</p> <p><a href="#">Sudo apt-get install whois</a></p> <pre> dpiche@dbrmd213_01:~/script\$ sudo apt-get install whois (sudo) password for dpiche: Reading package lists... Done Building dependency tree Reading state information... Done The following NEW packages will be installed:   whois 0 upgraded, 1 newly installed, 0 to remove and 66 not upgraded. Need to get 43.7 kB of archives. After this operation, 262 kB of additional disk space will be used. Get:1 http://archive.ubuntu.com/ubuntu bionic/main amd64 whois amd64 5.3.0 [43.7 kB] Fetched 43.7 kB in 0s (91.2 kB/s) Selecting previously unselected package whois. (Reading database ... 108919 files and directories currently installed.) Preparing to unpack .../archives/whois_5.3.0_amd64.deb ... Unpacking whois (5.3.0) ... Setting up whois (5.3.0) ... Processing triggers for man-db (2.8.3-2) ... dpiche@dbrmd213_01:~/script\$ </pre> <p>(Host On Net, 2014, page 1)</p> <p>2) Below is the main menu. If you select "Create-User" the file where the underlying function is stored will run and as a result users from a file will be created.</p>	



- 3) After every user that is created this message will pop up. A good thing is that there are only 3 users in the file. If there were 100 this screen will pop up 100 times.



- 4) Below is proof that the users were created from the file.



- 5) The screen below is a capture of the “/etc/passwd” file.

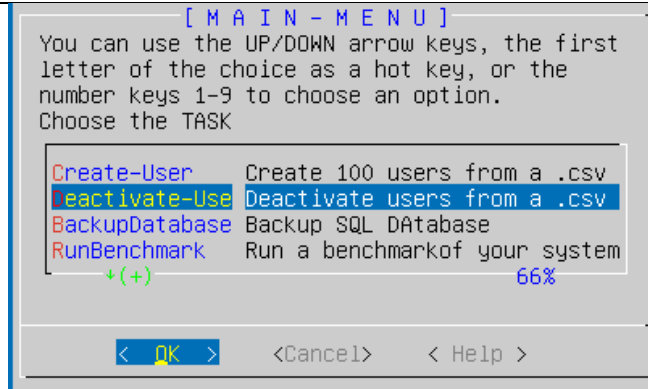


(Linux Questions, 2004, para. 13)

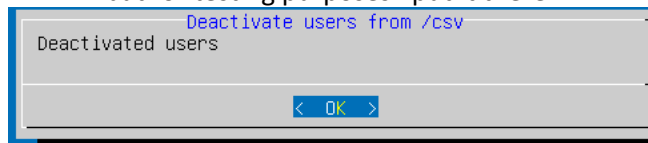
(Tec Mint, 2014, page 1)

(Stack Exchange, 2018, page 1)

- 6) This next option on the menu will delete the users that were created in the first step.



- 7) Again the Deactivate screen will pop up for every single user that is deactivated. You can remove it but for testing purposes I put it there.

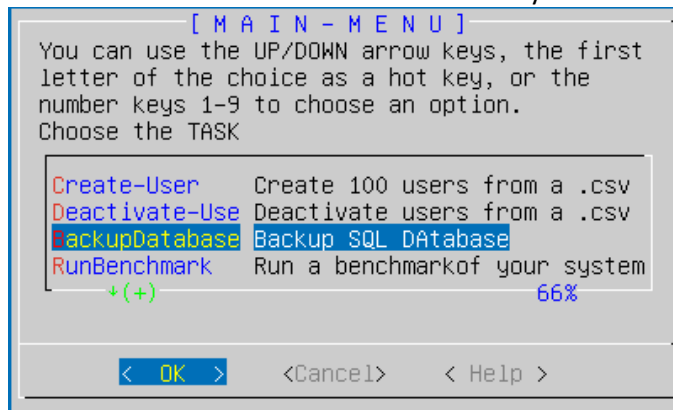


- 8) The screen below is the bottom of the “/etc/passwd” file. As you can see the three users “Bill, Dave, Daniel” are gone.

```
dpiche:x:1000:1000:Daniel Piche:/home/dpiche:/bin/bash
mysql:x:111:114:MySQL Server,,,:/nonexistent:/bin/false
apacheuser:x:1001:1001::/var/www:/bin/nologin
dpiche@dbcmd213_01:~/script$
```

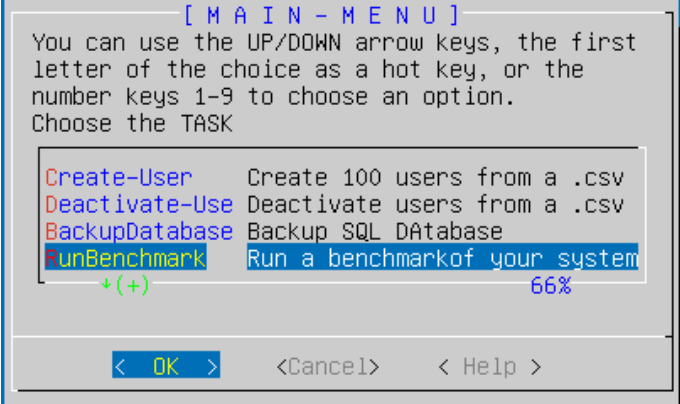
(Linux Questions, 2004, para. 13)

- 9) This next selection in the menu will backup a database. This is not a simple backup function. It checks to see if there is a match to the file it is trying to backup and if there is it will prompt you if you would like to overwrite the directory.



- 10) The screen below shows that the directory is not found and asks if you would like to copy it.

```
database Not Found!!
Would you like to copy /home/dpiche/script/database-06-11-10-15:28:31 to /home/dpiche/script/data(y
n)?y
[sudo] password for dpiche:
Copied directory
```

	<div>11) After hitting yes you can go to the directory and see that it is there with the date appended to the end of the directory name.</div> <div><pre>dpiche@dbbrmd213_01:~/script\$ cd data dpiche@dbbrmd213_01:~/script/data\$ ls database-01-11-18  database-02-11-18  database-03-11-18-19:25:25 dpiche@dbbrmd213_01:~/script/data\$ pwd /home/dpiche/script/data dpiche@dbbrmd213_01:~/script/data\$ _</pre></div> <div>12) If you run the backup again it will tell you that a directory with the same date has already been found.</div> <div>*Because I updated script to append time to folder name you will not get "Database Found"</div> <div><pre>database Not Found!! Would you like to copy /home/dpiche/script/database-06-11-18-15:29:56 to /home/dpiche/script/data(y/n)?n No directories copied</pre></div> <div>(Tec Admin, 2017, page 1)</div> <div>13) This next selection in the menu will run a benchmark of the system and output the results.</div> <div></div> <div>14) Below shows the output from 3 different benchmarks.</div>	
--	--	--

Run benchmark of system and display

```
run Benchmark of system
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
loop0 7:0 0 87.9M 1 loop /snap/core/5662
loop1 7:1 0 87.9M 1 loop /snap/core/5548
loop2 7:2 0 87.9M 1 loop /snap/core/5742
sda 8:0 0 50G 0 disk
├─sda1 8:1 0 1M 0 part
├─sda2 8:2 0 10G 0 part /home
└─sda3 8:3 0 40G 0 part /
sr0 11:0 1 1024M 0 rom
```

```
PID TTY TIME CMD
1569 tty1 00:00:00 bash
1695 tty1 00:00:00 bash
1714 tty1 00:00:00 dialog
3241 tty1 00:00:00 menu3.sh
3257 tty1 00:00:00 ps
```

```
MemTotal: 8144412 kB
MemFree: 7194156 kB
MemAvailable: 7514464 kB
Buffers: 39484 kB
Cached: 484296 kB
```

< OK >

15) Below shows the benchmark outputed to a file

```
dpiche@dbbrmd213_01:~/script$ cat performance
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
loop0 7:0 0 87.9M 1 loop /snap/core/5662
loop1 7:1 0 87.9M 1 loop /snap/core/5548
loop2 7:2 0 87.9M 1 loop /snap/core/5742
sda 8:0 0 50G 0 disk
├─sda1 8:1 0 1M 0 part
├─sda2 8:2 0 10G 0 part /home
└─sda3 8:3 0 40G 0 part /
sr0 11:0 1 1024M 0 rom
```

```
PID TTY TIME CMD
1695 tty1 00:00:00 bash
2000 tty1 00:00:00 menu3.sh
2006 tty1 00:00:00 ps
```

```
MemTotal: 8144420 kB
MemFree: 7363128 kB
MemAvailable: 7523212 kB
Buffers: 29276 kB
Cached: 338076 kB
```

```
dpiche@dbbrmd213_01:~/script$ pwd
/home/dpiche/script
dpiche@dbbrmd213_01:~/script$
```

(Linux Info, 2005, page 1)

(Nix Craft, 2013, page 1)

(PC Suggest, 2017, page 1)

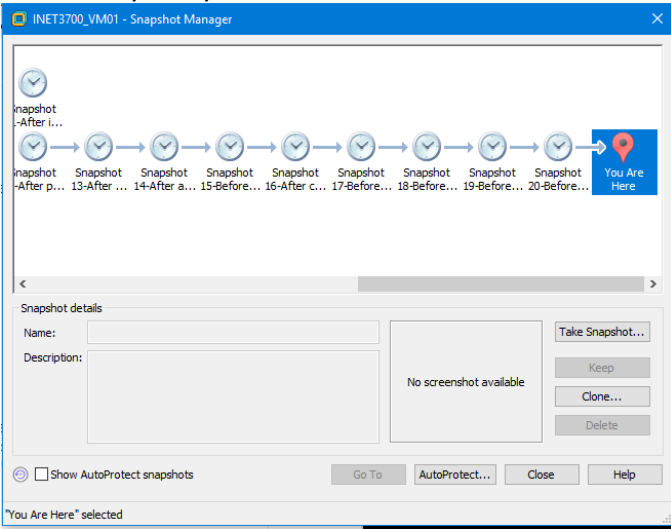
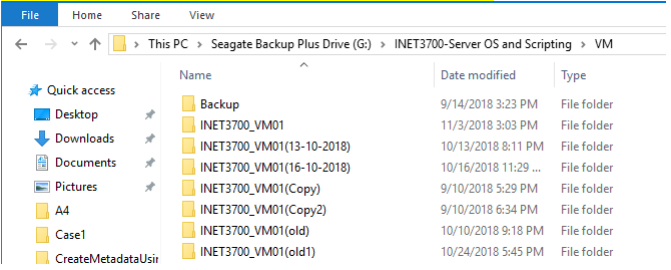
16) The next selection on the menu produces an update.



		<div data-bbox="493 189 1169 596" data-label="Image"> </div> <p>17) After selecting to do an update you will get the following output from the command prompt.</p> <div data-bbox="493 665 1169 854" data-label="Code-Block"> <pre>dpiche@dbrmd213_01:~/script\$ sudo apt update Hit:1 http://archive.ubuntu.com/ubuntu bionic InRelease Get:2 http://archive.ubuntu.com/ubuntu bionic-security InRelease [83.2 kB] Get:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB] Fetched 172 kB in 1s (205 kB/s) Reading package lists... Done Building dependency tree Reading state information... Done 66 packages can be upgraded. Run 'apt list --upgradable' to see them. dpiche@dbrmd213_01:~/script\$</pre> </div> <p>18) After the update you will be prompted</p> <div data-bbox="493 890 1169 1026" data-label="Image"> </div> <p>(Linux Config, 2018, page 1)</p> <p>19) This last selection to the menu will exit the menu.</p> <div data-bbox="493 1096 1136 1482" data-label="Image"> </div> <p>20) After selecting it you will be brought to the command prompt.</p> <div data-bbox="493 1554 972 1635" data-label="Code-Block"> <pre>Bye dpiche@dbrmd213_01:~/script\$</pre> </div> <p>(CyberCiti, 2016, page 1) See glossary for script</p>	
Task 2	Benchmark Script with crontab	<p>1) Below are 6 crontab entries. As you can see one of them is set to run “BenchScript.sh” every Monday at 12:01 am.</p> <p>2) To test that crontab and the script work I created a 5<sup>th</sup> crontab entry that ran the script today.</p>	

		<pre> # and day of week (dow) or use '*' in these fields (for 'any').# # Notice that tasks will be started based on the cron's system # daemon's notion of time and timezones. # # Output of the crontab jobs (including errors) is sent through # email to the user the crontab file belongs to (unless redirected). # # For example, you can run a backup of all your user accounts # at 5 a.m every week with: # 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/ # # For more information see the manual pages of crontab(5) and cron(8) # # m h dom mon dow   command 0 20 * * * /home/dpiche/script/CronScript.sh 0 20 * * 6 /home/dpiche/script/CronScript.sh 0 20 1 * 0 /home/dpiche/script/CronScript.sh 1 0 * * 1 /home/dpiche/script/BenchScript.sh  30 16 8 11 4 /home/dpiche/script/BenchScript.sh 30 16 8 11 4 /home/dpiche/script/CronScript.sh ~ ~ ~ ~ ~ ~ "crontab.K0VBX0/crontab" 30L, 1165C written crontab: installing new crontab </pre> <p>3) Below is the contents of the "BenchScript.sh" file. As you can see it appends 3 different benchmarks of the system to a file called "benchmark-\$date" located in the benchmark directory.</p> <pre> dpiche@dbbrmd213_01:~/script\$ cat BenchScript.sh #!/bin/bash  date=\$(date +%d-%m-%y-%T) directory=/home/dpiche/script/benchmark  echo \$date  touch \$directory/benchmark-\$date  lsblk &gt;&gt;\$directory/benchmark-\$date echo " " &gt;&gt; \$directory/benchmark-\$date ps &gt;&gt; \$directory/benchmark-\$date echo " " &gt;&gt; \$directory/benchmark-\$date head -5 /proc/meminfo &gt;&gt; \$directory/benchmark-\$date dpiche@dbbrmd213_01:~/script\$ pwd /home/dpiche/script dpiche@dbbrmd213_01:~/script\$ _ </pre> <p>4) When crontab reached the scheduled time to run I noticed a new file with the proper name and current date and time.</p> <pre> dpiche@dbbrmd213_01:~/script/benchmark\$ ls benchmark-03-11-18      benchmark-03-11-18-18:32:31  benchmark-08-11-18-16:27:01 benchmark-03-11-18-18:29:02  benchmark-03-11-18-18:37:24  benchmark-08-11-18-16:30:01 dpiche@dbbrmd213_01:~/script/benchmark\$ pwd /home/dpiche/script/benchmark </pre> <p>5) I opened the file with the current date and time that was created by the script and noticed that everything ran as expected.</p>	
--	--	--	--

		<pre> dpiche@dbcmd213_01:~/script/benchmark\$ ls benchmark-03-11-18      benchmark-03-11-18-18:32:31  benchmark-08-11-18-16:27:01 benchmark-03-11-18-18:29:02  benchmark-03-11-18-18:37:24  benchmark-08-11-18-16:30:01 dpiche@dbcmd213_01:~/script/benchmark\$ cat benchmark-08-11-18-16:30:01 NAME MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT loop0  7:0    0  87.9M  1 loop /snap/core/5742 loop1  7:1    0  87.9M  1 loop /snap/core/5662 loop2  7:2    0  87.9M  1 loop /snap/core/5548 sda    8:0    0   50G   0 disk ├─sda1  8:1    0    1M   0 part ├─sda2  8:2    0   10G   0 part /home ├─sda3  8:3    0   40G   0 part / sr0    11:0   1 1024M   0 rom    PID TTY          TIME CMD  1845 ?        00:00:00 systemd  1853 ?        00:00:00 (sd-pam)  2131 ?        00:00:00 sh  2133 ?        00:00:00 BenchScript.sh  2139 ?        00:00:00 ps  MemTotal:      8144420 kB MemFree:       7357600 kB MemAvailable:   7520272 kB Buffers:        29832 kB Cached:        340164 kB dpiche@dbcmd213_01:~/script/benchmark\$ </pre> <p>(David Both, 2017, page 1) (Stack Exchange, 2018, page 1) See glossary for script</p>	
	Crontab Backups	<ol style="list-style-type: none"> <li>1) The screen below shows the content of the crontab file.</li> <li>2) The three first entries are supposed to run the “CronScript.sh” script at three different times. Daily, weekly and monthly.</li> <li>3) To test that crontab works and so does the script I entered a sixth entry into crontab that will run the “CronScript.sh” file at the current date and time.</li> </ol> <pre> # and day of week (dow) or use '*' in these fields (for 'any').# # Notice that tasks will be started based on the cron's system # daemon's notion of time and timezones. # # Output of the crontab jobs (including errors) is sent through # email to the user the crontab file belongs to (unless redirected). # # For example, you can run a backup of all your user accounts # at 5 a.m every week with: # 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/ # # For more information see the manual pages of crontab(5) and cron(8) # # m h dom mon dow   command 0 20 * * * /home/dpiche/script/CronScript.sh 0 20 * * 6 /home/dpiche/script/CronScript.sh 0 20 1 * 0 /home/dpiche/script/CronScript.sh 1 0 * * 1 /home/dpiche/script/BenchScript.sh  30 16 8 11 4 /home/dpiche/script/BenchScript.sh 30 16 8 11 4 /home/dpiche/script/CronScript.sh .. .. .. .. .. "crontab.K0VBX0/crontab" 30L, 1165C written crontab: installing new crontab </pre> <ol style="list-style-type: none"> <li>4) As you can see, when the script ran it created a new entry in the data folder with the current date and time.</li> </ol> <pre> dpiche@dbcmd213_01:~/script/data\$ pwd /home/dpiche/script/data dpiche@dbcmd213_01:~/script/data\$ ls database-01-11-18  database-03-11-18-19:25:25  database-08-11-18-16:25:34 database-02-11-18  database-06-11-18-15:28:31  database-08-11-18-16:30:01 dpiche@dbcmd213_01:~/script/data\$ _ </pre> <ol style="list-style-type: none"> <li>5) Below is the scripts that ran with crontab.</li> </ol>	

		<pre>dpiche@dbbrmd213_01:~/script\$ cat CronScript.sh #!/bin/bash  date=\$(date +%d-%m-%Y-%T)  #echo \$date  #Backup Database daily every evening, weekly every friday evening, and monthly cp -R /home/dpiche/script/database /home/dpiche/script/data/database-\$date/ dpiche@dbbrmd213_01:~/script\$ pwd /home/dpiche/script dpiche@dbbrmd213_01:~/script\$</pre> <p>(David Both, 2017, page 1) (Stack Exchange, 2018, page 1)</p>	
Task 3	Snapshot	<p>1) Below shows a screen capture of the snapshot manager. It is important to take regular snapshots of your system.</p> 	
	Gold Copy	<p>1) I also recommend creating a gold copy of your system state every week. Below shows the path to get to the “Gold Copy”.</p> <p>2) I recommend storing this on a different drive so that if you misplace your drive you have an extra copy.</p> <p><b>G:\INET3700-Server OS and Scripting\VM</b></p> 	

## Summary

Writing scripts can be tricky because there are many variations of the syntax and every problem is different and calls for a different structure. Hopefully with the above documentation you can create a script from scratch and not have any issues.

## Glossary

### 1<sup>st</sup> Menu Box Script

```
#!/bin/bash

#Variables
INPUT=menuin2.sh.$$
OUTPUT=menuout2.sh.$$
vi_editor=${EDITOR-vi}

trap "rm $OUTPUT; rm $INPUT; exit" SIGHUP SIGINT SIGTERM

#Function that displays main output
function display_output() {
    local h=${1-10}
    local w=${2-41}
    local t=${3-Output}
    dialog --backtitle "Linux Script" --title "${t}" --clear --msgbox "${<$OUTPUT}" ${h} ${w}
}

#Function that shows current directory
function show_where() {
    echo "You are in the "$(pwd) " directory!" >$OUTPUT
    display_output 6 60 "Current Directory"
}

#Function that displays system details
function show_details() {
    echo "The owner of the current directory is : "$(whoami)>$OUTPUT
    echo " ">>$OUTPUT
    echo "Owner permissions are : "$(getfacl /home/dpiche/script) >>$OUTPUT
    echo " ">>$OUTPUT
    echo "The number of files in the current directory is : "$(ls -l | wc -l) >>$OUTPUT
    echo " ">>$OUTPUT
    str=$(ls -S | head -1)
    echo "The name and size of the biggest file is : "$(du -h $str) >>$OUTPUT
    echo " ">>$OUTPUT
    echo "The name of the biggest file is : "$(ls -S | head -1) >>$OUTPUT
    display_output 50 60 "Directory Info"
}

#While loop that displays and helps you select from menu
while true
do
    dialog --clear --help-button --backtitle "Linux Shell Script Assignment" \
```

```

--title "[ M A I N - M E N U]" \
--menu "You can use the UP/DOWN arrow keys, the first \n\
letter of the choice as a hot key, or the \n\
number keys 1-9 to choose an option. \n\
Choose the TASK" 15 50 4 \
Where "Where am I" \
DirDetails "Directory Details" \
Exit "Exit to the shell" 2>"${INPUT}"

menuitem=${<"${INPUT}"}

case $menuitem in
    Where) show_where;;
    DirDetails) show_details;;
    Exit) echo "Bye"; break;;
esac

done

[ -f $OUTPUT ] && rm $OUTPUT
[ -f $INPUT ] && rm $INPUT

```

## 2<sup>nd</sup> Menu Box Script

```

#!/bin/bash

#Variables
INPUT=menuin3.sh.$$
OUTPUT=menuout3.sh.$$
vi_editor=${EDITOR-vi}
PERFORMANCE=performance

trap "rm $OUTPUT; rm $INPUT; exit" SIGHUP SIGINT SIGTERM

#Function that displays main menu
function display_output() {
    local h=${1-10}
    local w=${2-41}
    local t=${3-Output}
    dialog --backtitle "Linux script" --title "${t}" --clear --msgbox "${<$OUTPUT}" ${h} ${w}
}

#Function that deactivates users
function Deactivate() {
    awk -F"," '{print $1, $2}' A4.csv | while read UN PW
do
    sudo userdel $UN
    echo "Deactivated users" >$OUTPUT
done

```

```

        display_output 6 60 "Deactivate users from /csv"
done
}

#Function that creates users
function Create_User() {
    awk -F"," '{print $1, $2}' A4.csv | while read UN PW
do
    sudo useradd -p $PW $UN
    echo "Created users">$OUTPUT
    display_output 6 60 "Create users from .csv"
done

}

#Function that creates a backup
function Backup() {
    database=/home/dpiche/script/database
    backup=/home/dpiche/script/data
    date=$(date +%d-%m-%y-%T)

    if test -d $backup/database-$date;
    then
        echo "Found database!!!"
    else echo "database Not Found!!"
    fi

    echo -n "Would you like to copy $database-$date to $backup(y/n)?"
    read answer
    if echo "$answer" | grep -iq "^y" ;then
        sudo cp -R $database $backup/database-$date/
        echo "Copied directory"
        read value
    else
        echo "No directories copied"
        read value
    fi

}

#Function that gets a benchmark of the system
function Benchmark() {
    echo "run Benchmark of system">$OUTPUT
    echo " ">$PERFORMANCE
    lsblk " ">> $PERFORMANCE
    lsblk>>$OUTPUT
    echo " ">>$PERFORMANCE
    echo " ">>$OUTPUT
}

```

```

ps>>$PERFORMANCE
ps>>$OUTPUT
echo " ">>$PERFORMANCE
echo " ">>$OUTPUT
head -5 /proc/meminfo>>$PERFORMANCE
head -5 /proc/meminfo>>$OUTPUT
echo " ">>$PERFORMANCE
echo " ">>$OUTPUT
display_output 60 60 "Run benchmark of system and display"
}

```

#### #Function that updates the system

```

function Update() {
    sudo apt update
    echo "Updated System">>$OUTPUT
    display_output 6 60 "Update of system"
}

```

#### #While loop that displays the menu and allows you to select from it

```

while true
do

dialog --clear --help-button --backtitle "Linux shell script number 2" \
--title "[ M A I N - M E N U ]" \
--menu "You can use the UP/DOWN arrow keys, the first \n\
letter of the choice as a hot key, or the \n\
number keys 1-9 to choose an option. \n\
Choose the TASK" 15 50 4 \
Create-User "Create 100 users from a .csv file" \
Deactivate-User "Deactivate users from a .csv file" \
BackupDatabase "Backup SQL DAtabase" \
RunBenchmark "Run a benchmarkof your system" \
UpdateSystem "Update your system" \
Exit "Exit to the shell" 2>"${INPUT}"

menuitem=${<"${INPUT}"}

case $menuitem in
    Create-User) Create_User;;
    Deactivate-User) Deactivate;;
    BackupDatabase) Backup;;
    RunBenchmark) Benchmark;;
    UpdateSystem) Update;;
    Exit) echo "Bye"; break;;
esac
done

[ -f $OUTPUT ] && rm $OUTPUT

```



```
[ -f $INPUT ] && rm $INPUT
```

### Benchmark Script with crontab

```
#!/bin/bash

#Variables
date=$(date +%d-%m-%y-%T)
directory=/home/dpiche/script/benchmark

echo $date

touch $directory/benchmark-$date

#Block that prints benchmark info to a file and to screen
lsblk >>$directory/benchmark-$date
echo " " >> $directory/benchmark-$date
ps >> $directory/benchmark-$date
echo " " >> $directory/benchmark-$date
head -5 /proc/meminfo >> $directory/benchmark-$date
```

## Change Management Log

Below are all the changes made to the virtual machine

ChangeID	UserName	Role	DateOfChan	TimeOfChan	ChangeMade	DescriptionCommand
1	root	Administrator	10/24/2018	5:52:00 PM	Created a gold copy	See external drive on /INET3700 directory
2	root	Administrator	10/24/2018	5:59:00 PM	Create snapshot	Before begin assignment
3	root	Administrator	10/24/2018	7:00:00 PM	Install menuboxes	sudo apt-get install dialog
4	dpiche	Administrator	10/24/2018	7:14:00 PM	Create files	touch menu2.sh touch menuout2.sh touch menuin2.sh
5	dpiche	Administrator	10/24/2018	7:20:00 PM	Allow execute	chmod +x menu2.sh
6	dpiche	Administrator	10/24/2018	8:01:00 PM	Edit script	vi menu2.sh
7	dpiche	Administrator	10/26/2018	4:20:00 PM	Create files	touch menu3.sh touch menuin3.sh touch menuout3.sh
8	dpiche	Administrator	10/26/2018	4:50:00 PM	Allow execute	chmod +x menu3.sh
9	dpiche	Administrator	10/28/2018	3:15:00 PM	Edit script	vi menu3.sh
10	dpiche	Administrator	10/28/2018	3:20:00 PM	create backup directory	mkdir data mkdir database
11	dpiche	Administrator	11/3/2018	4:01:00 PM	Create benchmark script	touch BenchScript.sh
12	dpiche	Administrator	11/3/2018	4:10:00 PM	Allow execute	chmod +x BenchScript.sh
13	dpiche	Administrator	11/3/2018	4:13:00 PM	Edit script	vi BenchScript.sh
14	dpiche	Administrator	11/3/2018	4:15:00 PM	make vim editor	export EDITOR=vim
15	dpiche	Administrator	11/3/2018	4:16:00 PM	edit crontab	crontab -e
16	dpiche	Administrator	11/3/2018	4:31:00 PM	Create crontab backup script	touch CronScript.sh
17	dpiche	Administrator	11/3/2018	4:32:00 PM	Allow execute	chmod +x CronScript.sh
18	dpiche	Administrator	11/3/2018	4:33:00 PM	Edit script	vi CronScript.sh
19	dpiche	Administrator	11/3/2018	4:34:00 PM	edit crontab	crontab -e
20	dpiche	Administrator	11/9/2018	10:08:00 AM	Created a gold copy	See external drive on /INET3700 directory
21	dpiche	Administrator	11/9/2018	10:09:00 AM	Create snapshot	End of Assignment
* (New)						



INET3700\_VM01\_A4-  
2.pdf

## References

CyberCiti. (2016, March 29). A menu Box. Retrieved from [https://bash.cyberciti.biz/guide/A\\_menu\\_box](https://bash.cyberciti.biz/guide/A_menu_box)

StackOverflow. (2018). How can I see the size of files and directories in Linux. Retrieved from <https://stackoverflow.com/questions/11720079/how-can-i-see-the-size-of-files-and-directories-in-linux>

StackExchange. (2018). How to find the largest file in a directory. Retrieved from <https://superuser.com/questions/228853/how-to-find-the-largest-file-in-a-directory>

Host On Net. (2014, October 14). How to install whois client on Ubuntu. Retrieved from <https://blog.hostonnet.com/how-to-install-whois-client-in-ubuntu>

Linux Questions. (2004, August). Alternative Bash Script. Retrieved from <https://www.linuxquestions.org/questions/linux-server-73/bulk-user-add-script-693777/>

Tec Mint. (2014, March 28). The complete guide to useradd Command in Linux. Retrieved from <https://www.tecmint.com/add-users-in-linux/>

Tec Admin. (2017, March 22). How to get current date and time in Bash. Retrieved from <https://tecadmin.net/get-current-date-and-time-in-bash/>

Stack Exchange. (2018, November 5). Creating users from a .csv file. Retrieved from <https://unix.stackexchange.com/questions/414361/creating-users-from-a-csv-file>

Linux Info. (2005). The PS command. Retrieved from <http://www.linfo.org/ps.html>

Nix Craft. (2013, December 28). Find RAM size in Linux. Retrieved from <https://www.cyberciti.biz/faq/ram-size-linux/>

Nix Craft. (2013, June 9). Linux file size command. Retrieved from <https://www.cyberciti.biz/faq/how-to-see-file-size-on-linux-unix/>

PC Suggest. (2017, December 7). Benchmarking Linux Systems with Command Line tools. Retrieved from <https://www.pcsuggest.com/benchmarking-linux-systems-with-simple-command-line-tools/>

Linux Config. (2018, December 3). How to Install Ubuntu packages on Bionic Beaver Linux. Retrieved from <https://linuxconfig.org/how-to-update-ubuntu-packages-on-18-04-bionic-beaver-linux>

David Both. (2017, November 6). How to use cron in Linux. Retrieved from <https://opensource.com/article/17/11/how-use-cron-linux>

Stack Exchange, (2018, November 11). How can I set “vi” to be my default editor in Unix. Retrieved from <https://unix.stackexchange.com/questions/73484/how-can-i-set-vi-as-my-default-editor-in-unix>

## Helpful Links

how to create variables

<http://tldp.org/HOWTO/Bash-Prog-Intro-HOWTO-5.html>

export commands

[https://www.tutorialspoint.com/unix\\_commands/export.htm](https://www.tutorialspoint.com/unix_commands/export.htm)