OLD DOMINION UNIVERSITY

CYSE 270 LINUX SYSTEM FOR CYBERSECURITY

Assignment #9 Shell ScriptingWorking with Directory

John Wilson

01179411

CYSE 270 Assignment #9 Shell ScriptingWorking with Directory

Task A – Check files

Write a script like below that gets a list of files in a directory you entered, plus your name. Your script should pop out an error if an invalid directory is entered.
You need to test the following directories with your script:

- /etc/systemd
- /home
- A directory that does not exist (see PDF for screen shot example)

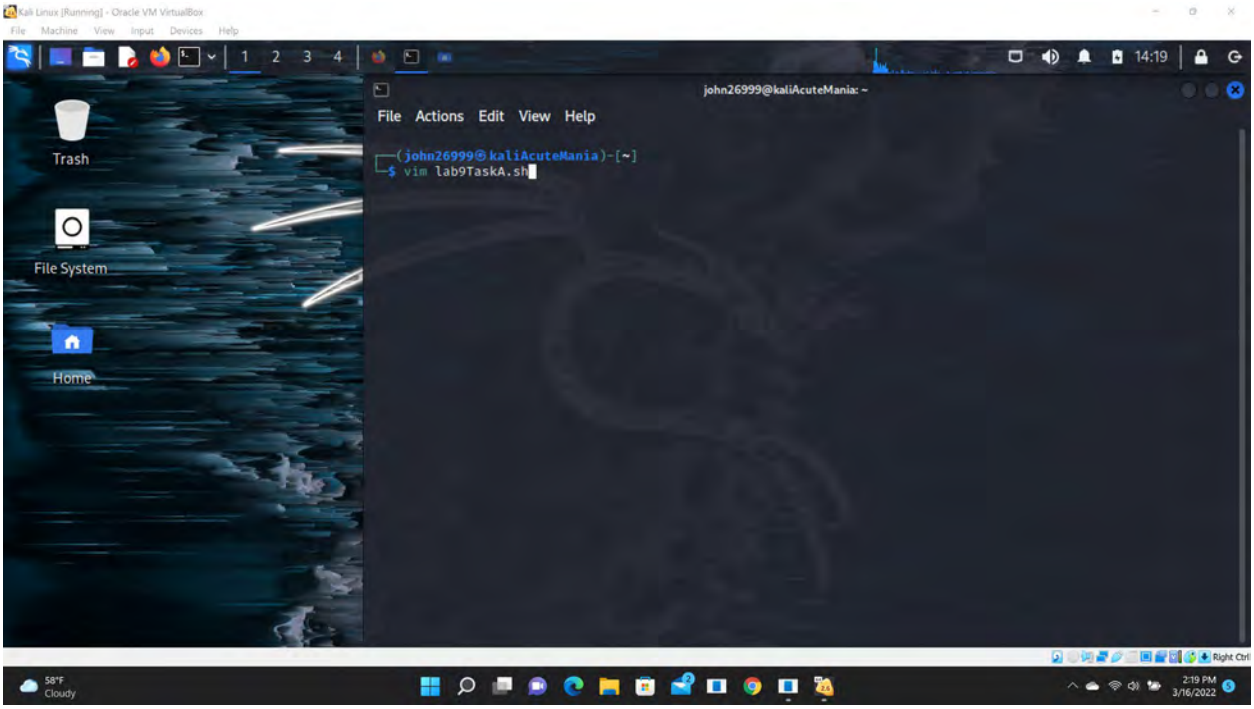
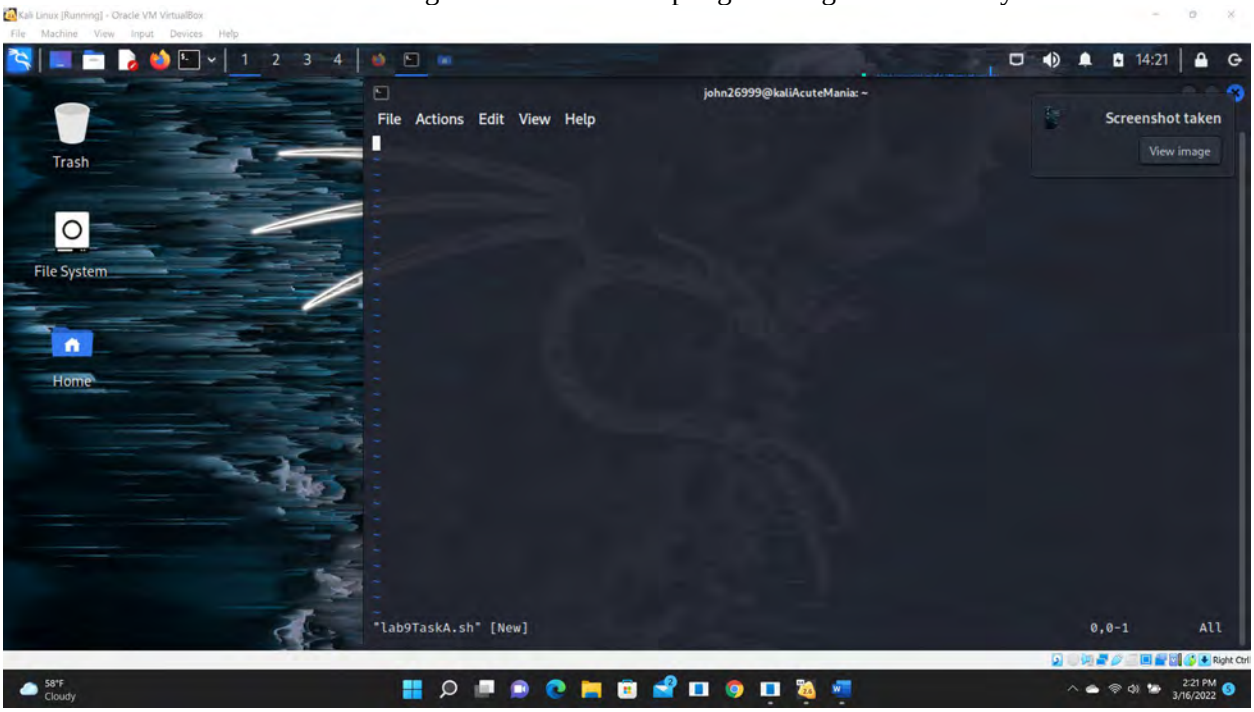
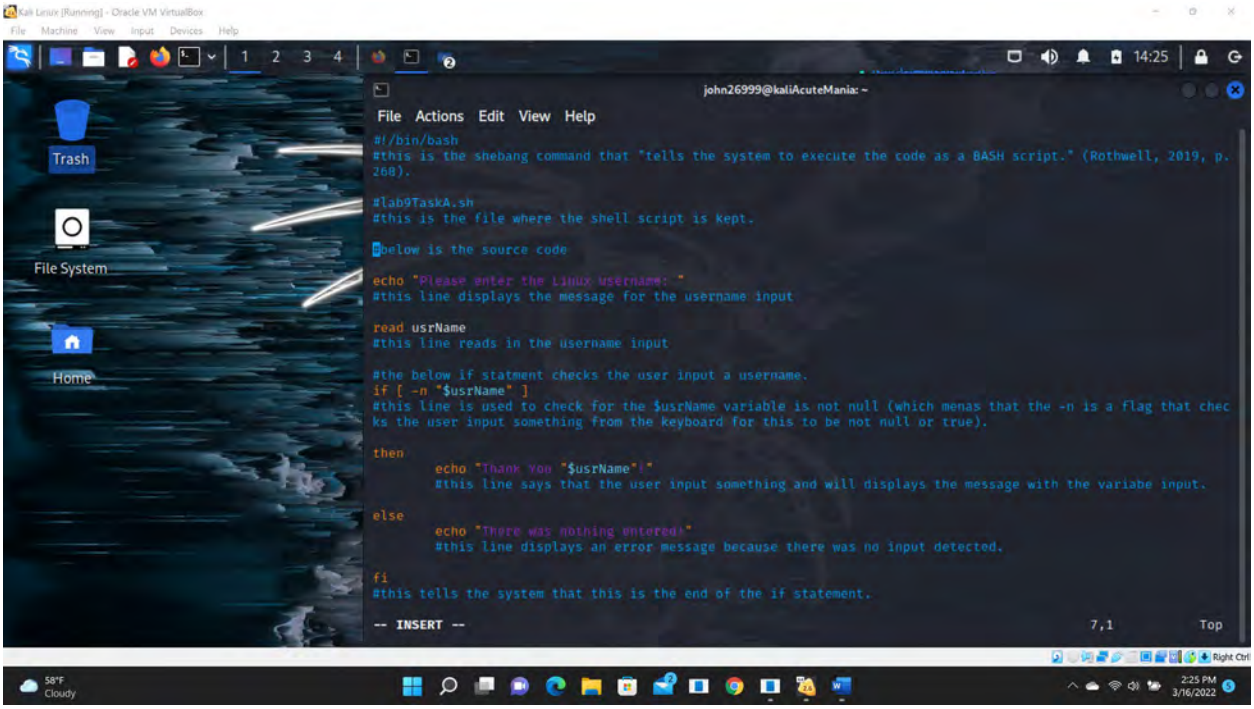


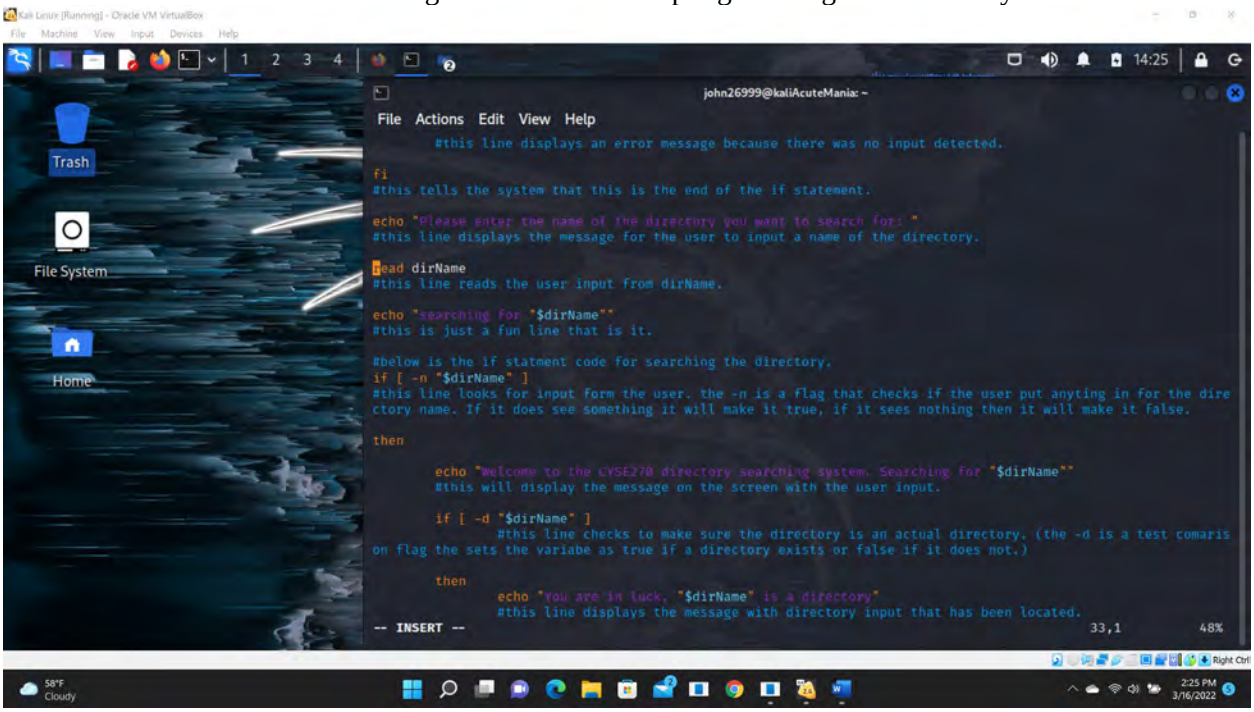
Figure 1 Screenshots of JWILS082 Computer screen for Step 1.

Above is the screen shot using the commands “vim lab9TaskA.sh” which opens up the VIM editor so that you can place the BASH commands in the document. “vim” is the command to open up the VIM editor. “lab9TaskA.sh” is the name of the file you want the vim editor to open or create (if not already created) and open.

CYSE 270 Assignment #9 Shell ScriptingWorking with Directory







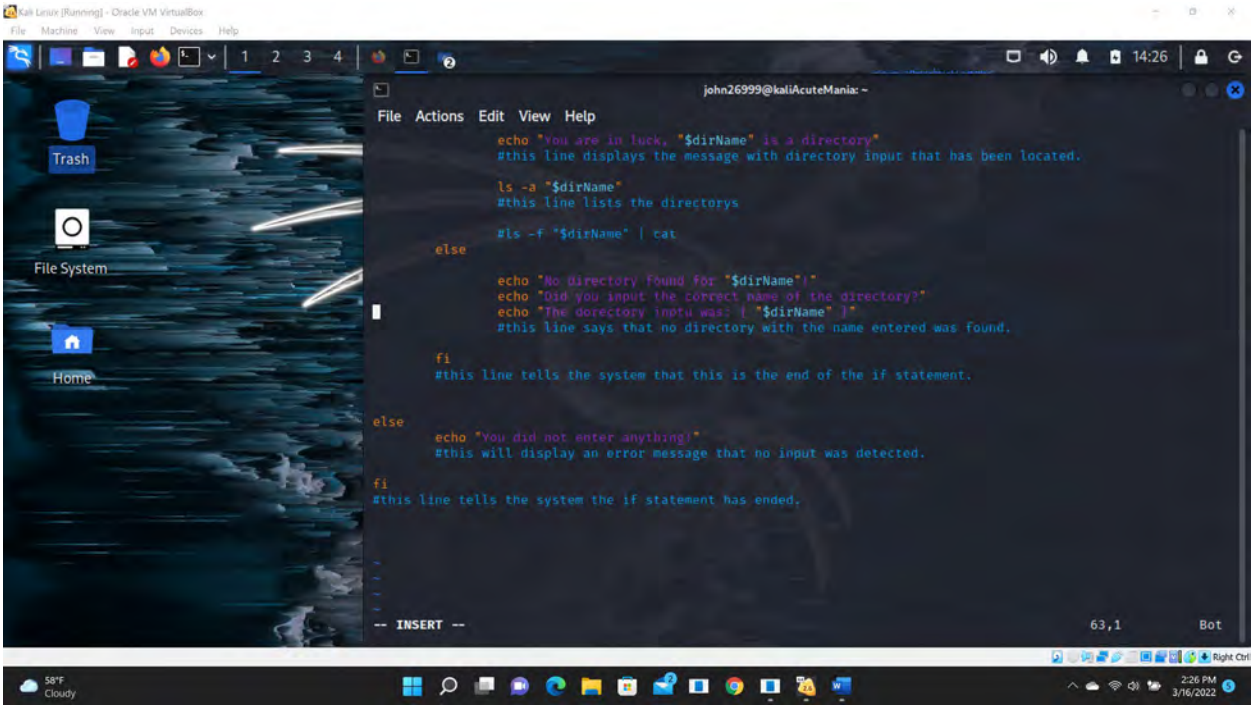


Figure 2 Screenshots of JWILS082 Computer screen for Step 2.

Above is the screen shot of the VIM editor open the new document lab9TaskA.sh.

The second and following shots are of the code that was put into the document using the VIM editor which opens up the vim editor so that you can place the BASH commands in the document. “vim” is the command to open up the vim editor. “lab9TaskA.sh” is the name of the file you want the vim editor to open or create (if not already created) and open.

CYSE 270 Assignment #9 Shell ScriptingWorking with Directory

Below is the actual code used with comments for Task A in the case the screen capture did not do a good enough job.

```
#!/bin/bash
#this is the shebang command that "tells the system to execute the code as a
BASH script." (Rothwell, 2019, p.268).

#lab9TaskA.sh
#this is the file where the shell script is kept.

#below is the source code

echo "Please enter the Linux username: "
#this line displays the message for the username input

read usrName
#this line reads in the username input

#the below if statment checks the user input a username.
if [ -n "$usrName" ]
#this line is used to check that the $usrName variable is not null (or empty)
(which menas that the -n is a flag that checks the user input something from
the keyboard for this to be not null or true).

then
    echo "Thank You "$usrName"!"
    #this line says that the user input something and will displays the
message with the variabe input.

else
    echo "There was nothing entered!"
        #this line displays an error message because there was no input
detected.

fi
#this tells the system that this is the end of the if statement.

echo "Please enter the name of the directory you want to search for: "
#this line displays the message for the user to input a name of the
directory.

read dirName
#this line reads the user input from dirName.

echo "searching for "$dirName""
#this is just a fun line that is it.

#below is the if statment code for searching the directory.
if [ -n "$dirName" ]
#this line looks for input form the user. the -n is a flag that checks if the
user put anyting in for the directory name. If it does see something it will
make it true, if it sees nothing then it will make it false.

then
```

CYSE 270 Assignment #9 Shell ScriptingWorking with Directory

```
    echo "Welcome to the CYSE270 directory searching system. Searching for
"$dirName""
    #this will display the message on the screen with the user input.

    if [ -d "$dirName" ]
        #this line checks to make sure the directory is an actual directory.
(the -d is a test comarison flag the sets the variabe as true if a directory
exists or false if it does not.)

    then
        echo "You are in luck, "$dirName" is a directory"
        #this line displays the message with directory input that has been
located.

        ls -a "$dirName"
        #this line lists the directorys

    else
        echo "No directory found for "$dirName"!"
        echo "Did you input the correct name of the directory?"
        echo "The directory input was: { "$dirName" }"
        #this line says that no directory with the name entered was found.

    fi
    #this line tells the system that this is the end of the if statement.


else
    echo "You did not enter anything!"
    #this will display an error message that no input was detected.

fi
#this line tells the system the if statement has ended.
```

CYSE 270 Assignment #9 Shell ScriptingWorking with Directory

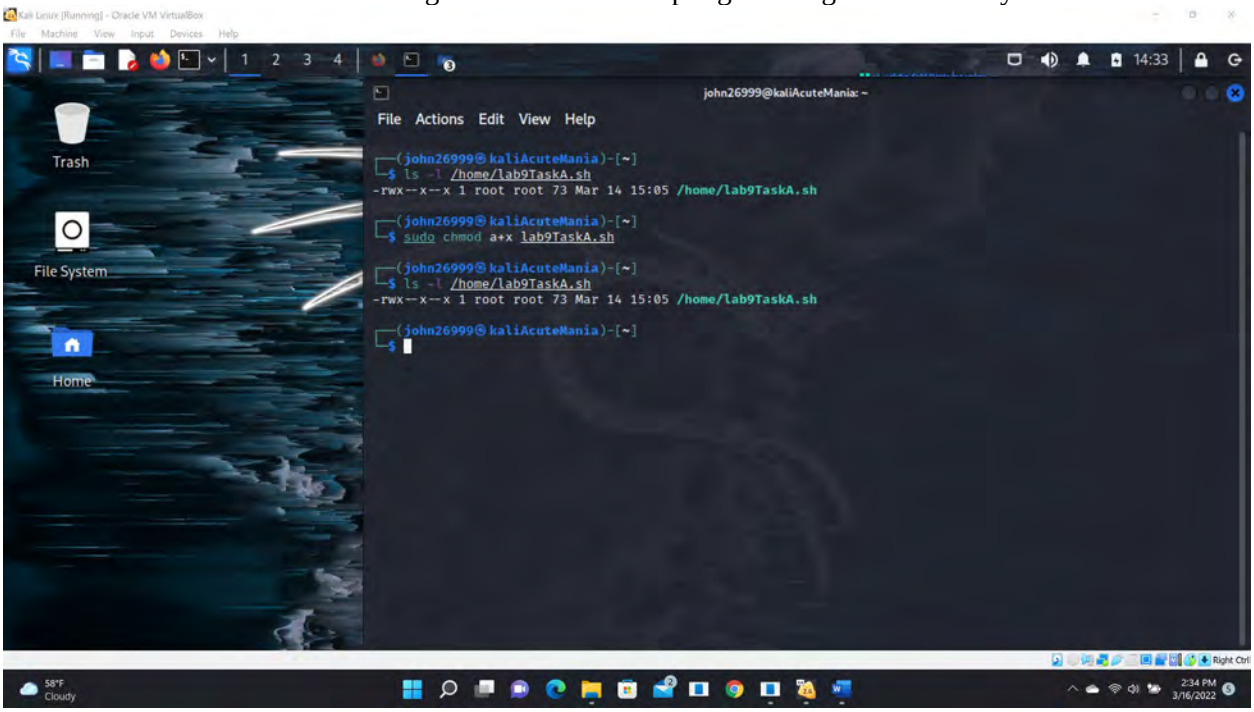


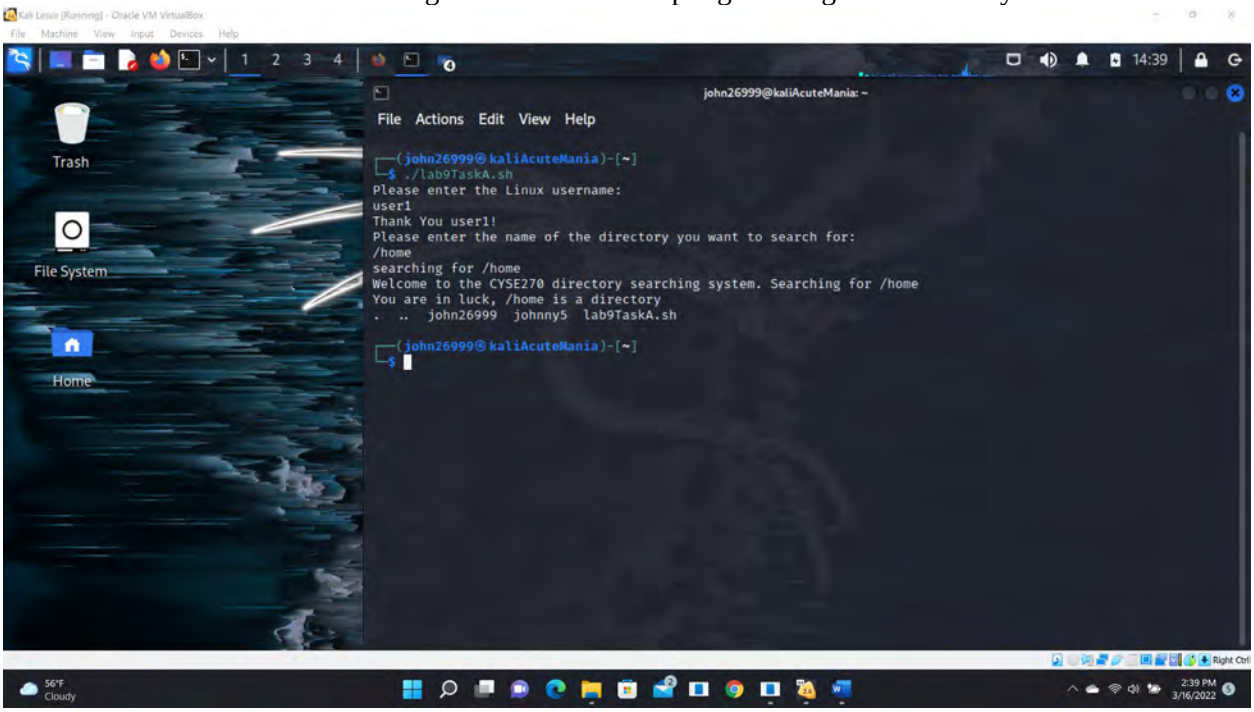
Figure 3 Screeshots of JWILS082 Computer screen for Step 3.

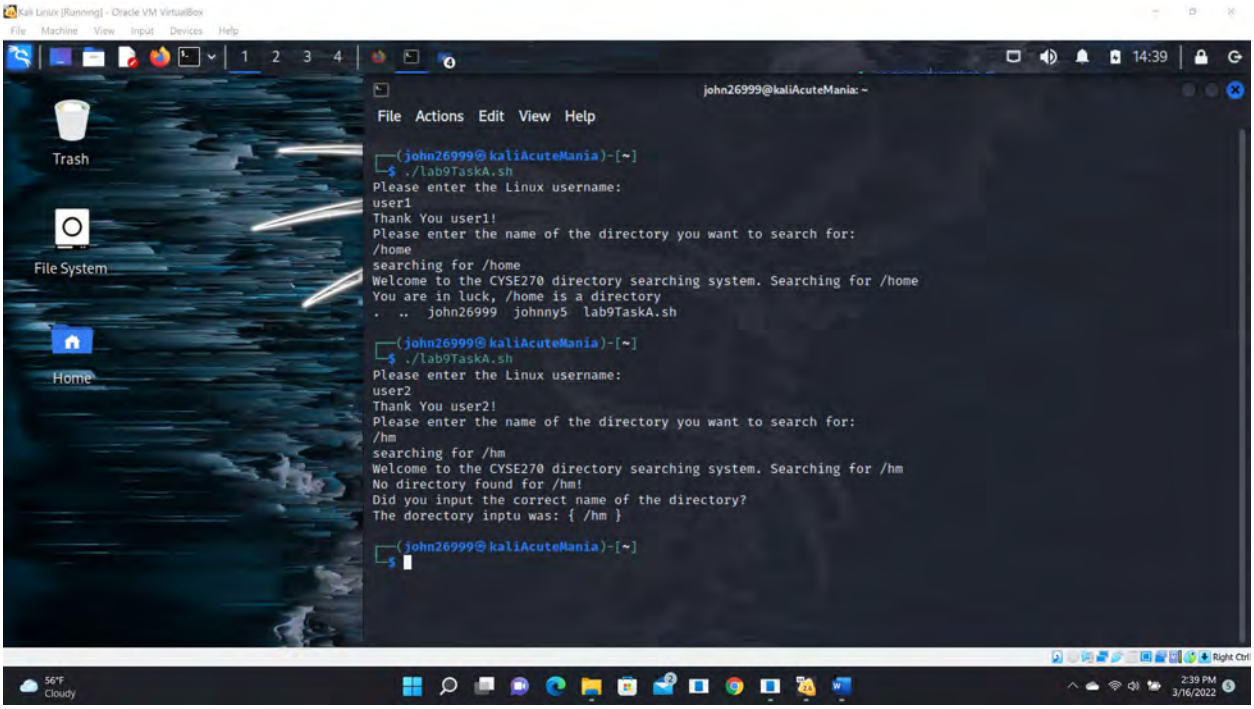
Above is the screen shot used the commands “ls -l lab9TaskA.sh” that shows the current permissions for the file labpTaskA.sh. “ls” is the command that lists the files. “-l” is the command that says to show all the file information. And lab9TaskA.sh is the name of the file.

I also using the commands “sudo chmod a+x lab9TaskA.sh” that updates the permissons to the file so that the BASH code can be executed.

Then I checked the file by using the command “ls -l lab9TaskA.sh” to validate the permissions were updated so that I can execute the script.

CYSE 270 Assignment #9 Shell ScriptingWorking with Directory





CYSE 270 Assignment #9 Shell ScriptingWorking with Directory

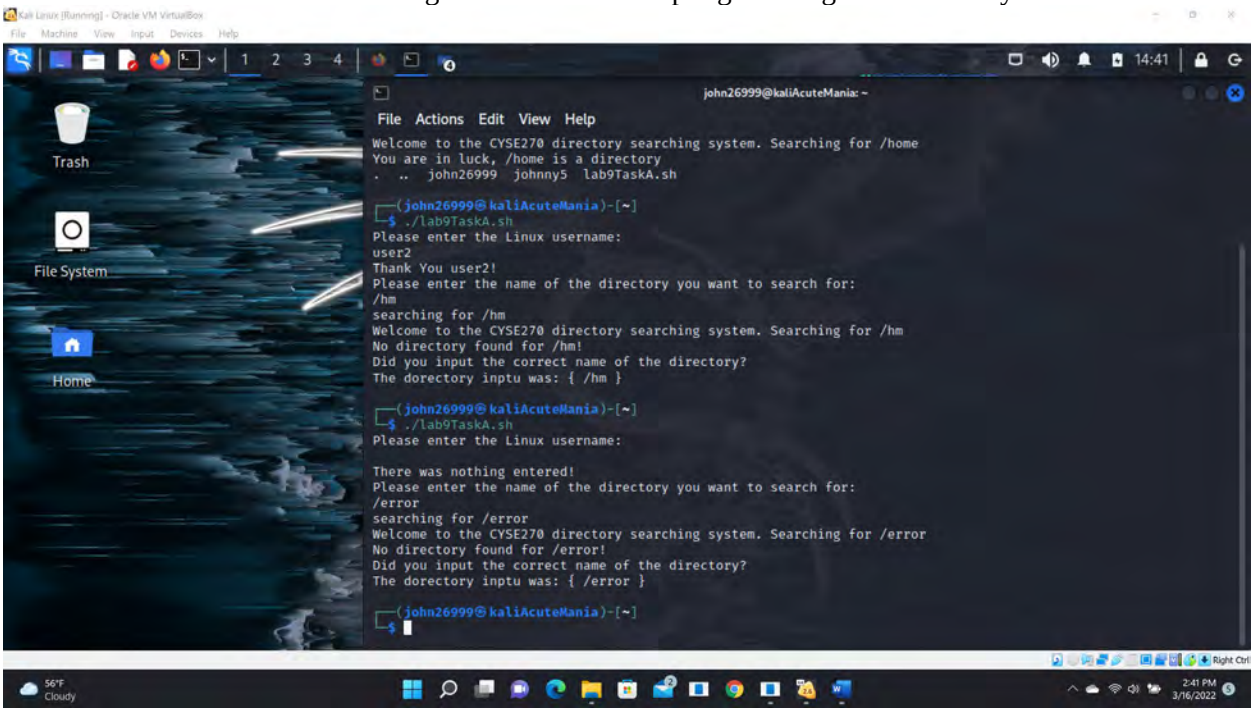


Figure 4 Screeshots of JWILS082 Computer screen for Step 4.

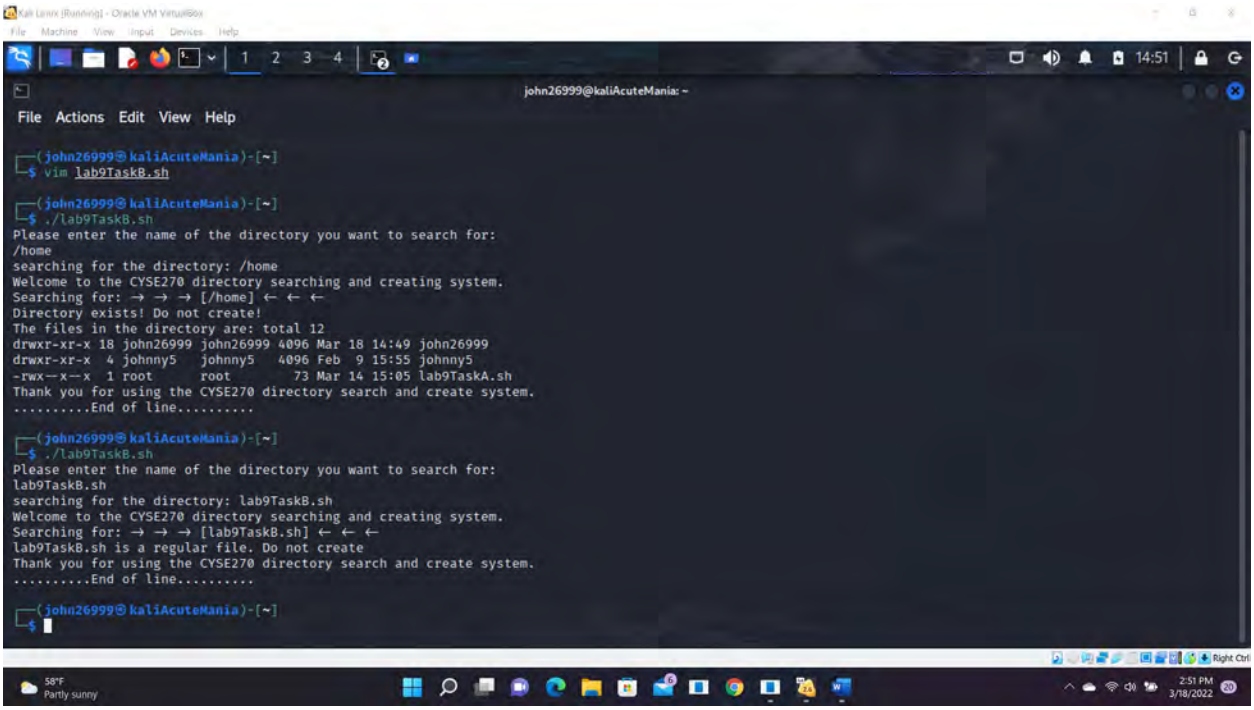
Above is the screen shot used to execute the script that gets a list of files in a directory you entered, plus your name. Your script should pop out an error if an invalid directory is entered.

CYSE 270 Assignment #9 Shell ScriptingWorking with Directory

Task B – Create a new directory

Write a shell script like below, that performs the following task:

1. Reads the name of the directory.
2. Check whether the given input is a directory or regular file.
3. If the input is a directory and it exists, then display the message “Directory exists. Do not create”.
4. If the input is a regular file, then display the message “Directory exists. Do not create”.
5. If the given input name in step-1 doesn’t exist, then create the new directory with the given name in step-1.



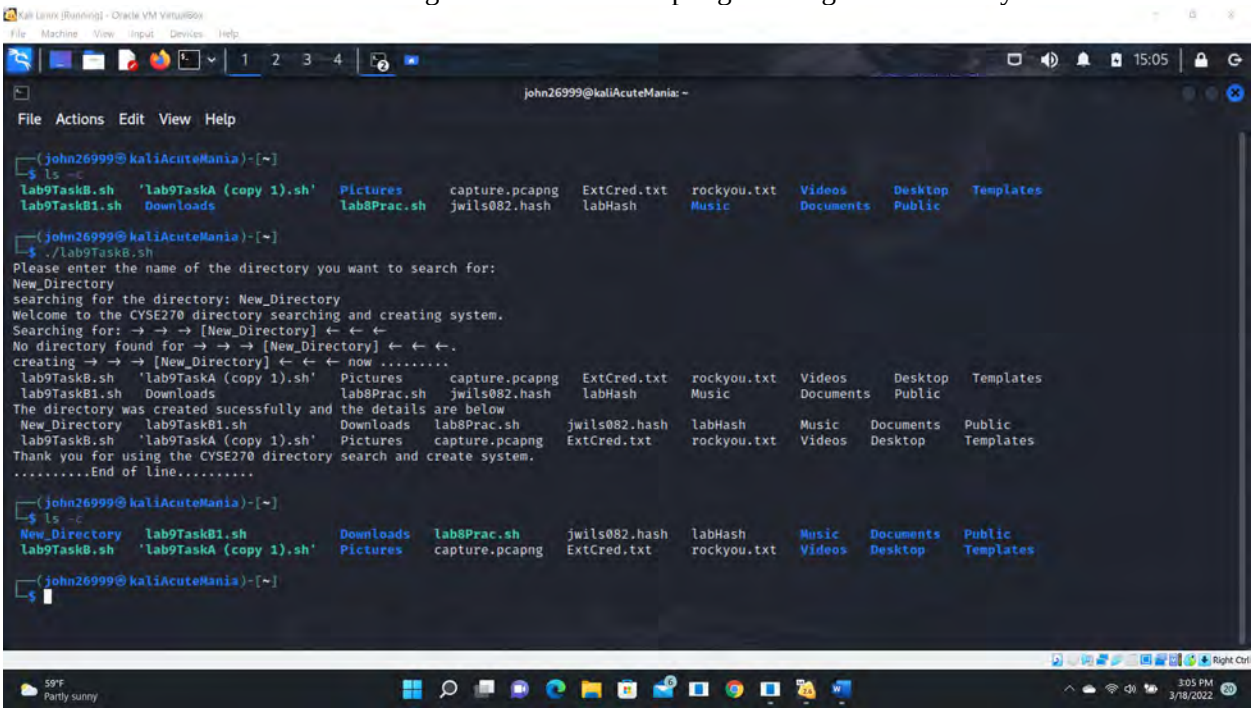
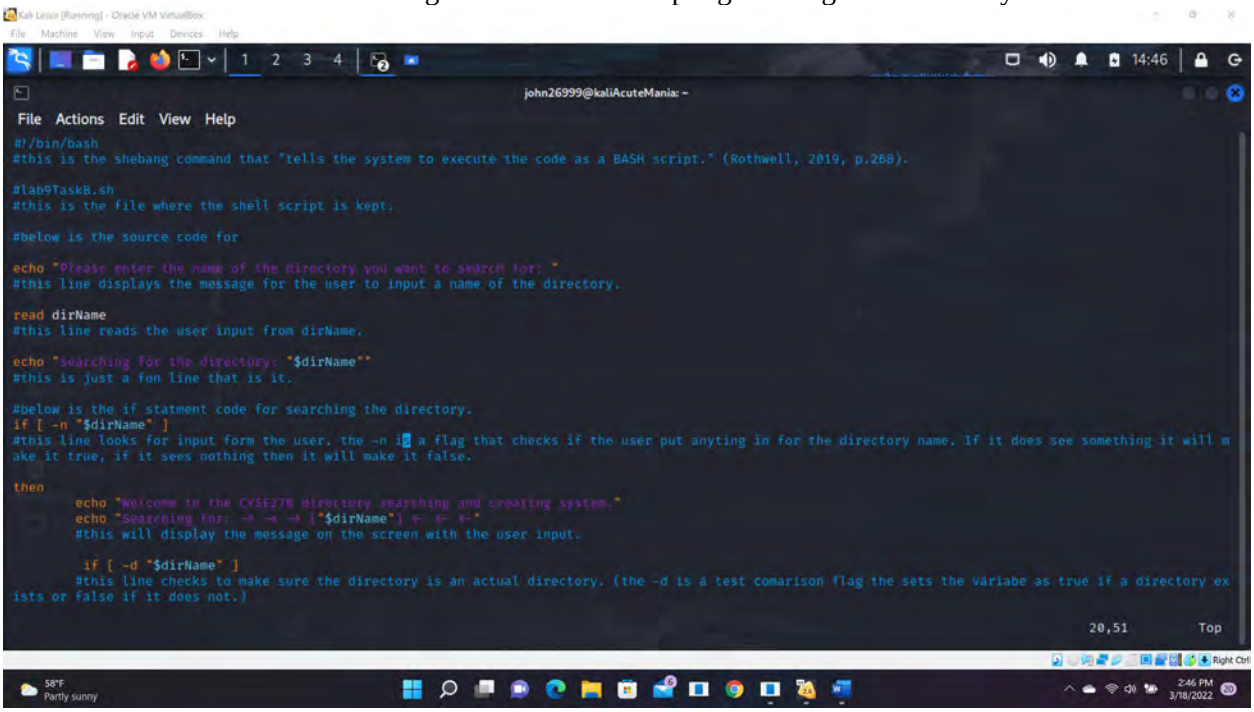


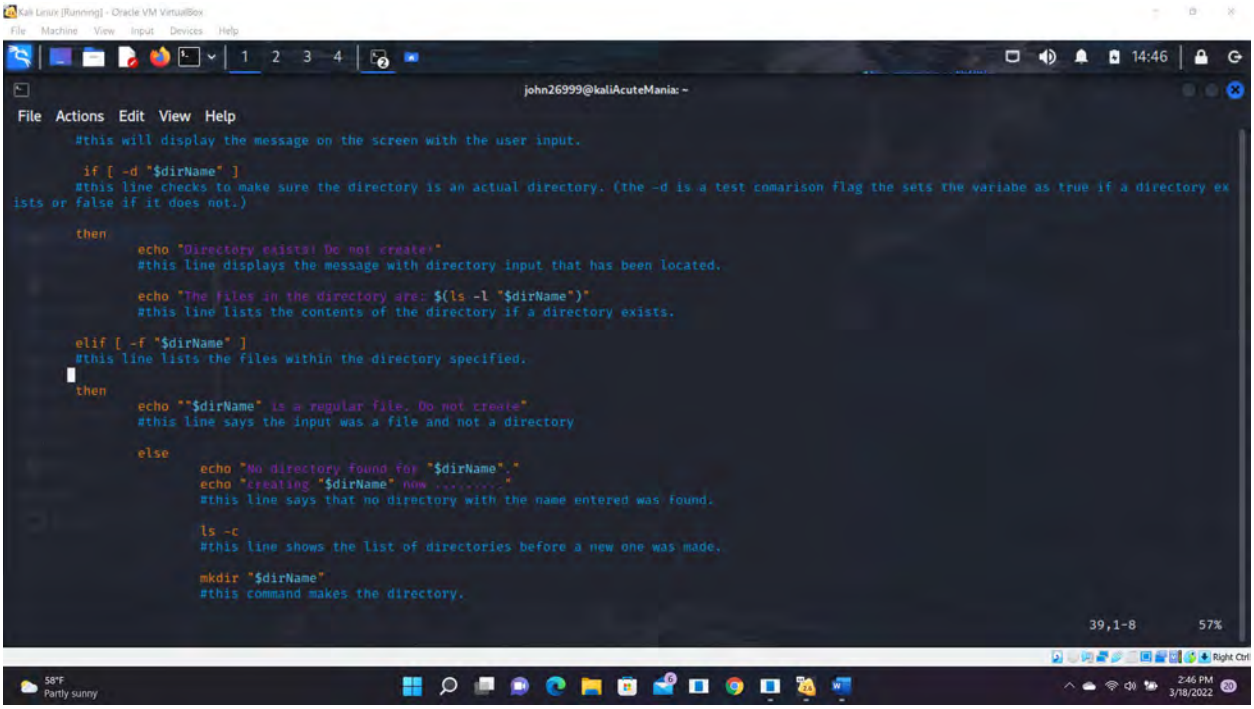
Figure 5 Screeshots of JWILS082 Computer screen for Steps of TaskB.

The first screenshot illiustrates that if a directory was used that already exists it will tell you and will not automatically create a new directory. In addition, I also entered a file that already existed to show that the script would detect the file and will not create it automatically.

The second screenshot is searching for a directory that does not exist and will automatically create it. In addition, it should tell the user the directory was not discovered, will create it automatically, and should show the directory created after its creation. The directory created is called “New_Directory” because it is literally a new directory.

CYSE 270 Assignment #9 Shell ScriptingWorking with Directory





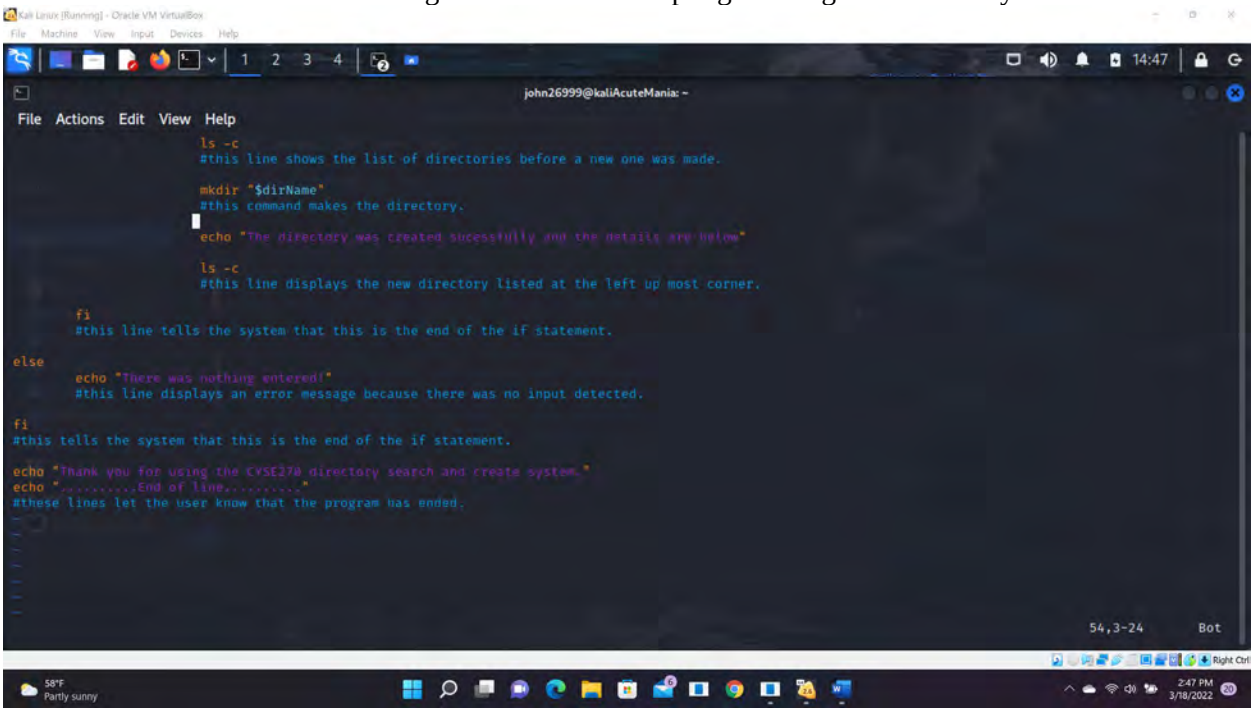


Figure 6 Screeshots of JWILS082 Computer screen of the code in VIM for TaskB..

Above are the screen shots of the code in VIM for TASK B.

Below is the actual code with comments written in the case the screenshots are too small to read.

```
#!/bin/bash
#this is the shebang command that "tells the system to execute the code as a
BASH script." (Rothwell, 2019, p.268).

#lab9TaskB.sh
#this is the file where the shell script is kept.

#below is the source code for

echo "Please enter the name of the directory you want to search for: "
#this line displays the message for the user to input a name of the
directory.

read dirName
#this line reads the user input from dirName.

echo "searching for the directory: "$dirName""
#this is just a fun line that is it.

#below is the if statment code for searching the directory.
if [ -n "$dirName" ]
#this line looks for input form the user. the -n is a flag that checks if the
user put anyting in for the directory name. If it does see something it will
make it true, if it sees nothing then it will make it false.

then
    echo "Welcome to the CYSE270 directory searching and creating system."
```

CYSE 270 Assignment #9 Shell ScriptingWorking with Directory

```
    echo "Searching for: -> -> -> ["$dirName"] <- <- <-"
    #this will display the message on the screen with the user input.

    if [ -d "$dirName" ]
    #this line checks to make sure the directory is an actual directory. (the
-d is a test comarison flag the sets the variabe as true if a directory
exists or false if it does not.)

    then
        echo "Directory exists! Do not create!"
        #this line displays the message with directory input that has been
located.

        echo "The files in the directory are: $(ls -l "$dirName")"
        #this line lists the contents of the directory if a directory exists.

    elif [ -f "$dirName" ]
    #this line lists the files within the directory specified.

    then
        echo ""$dirName" is a regular file. Do not create"
        #this line says the input was a file and not a directory

        else
            echo "No directory found for -> -> -> ["$dirName"] <- <- <-."
            echo "creating -> -> -> ["$dirName"] <- <- <- now ........."
            #this line says that no directory with the name entered was
found.

                ls -c
            #this line shows the list of directories before a new one was
made.

            mkdir "$dirName"
            #this command makes the directory.

            echo "The directory was created sucessfully and the details are
below"

            ls -c
            #this line displays the new directory listed at the left up most
corner.

    fi
    #this line tells the system that this is the end of the if statement.

else
    echo "There was nothing entered!"
        #this line displays an error message because there was no input
detected.

fi
#this tells the system that this is the end of the if statement.

echo "Thank you for using the CYSE270 directory search and create system."
echo ".........End of line........."
#these lines let the user know that the program has ended.
```

JOHN WILSON
3/21/22

CYSE 270 Assignment #9 Shell ScriptingWorking with Directory

Please note: Each student needs to screenshot your source code, as well as add necessary explanations to the core parts, for example, the logic behind the code, and the design of the control flow, etc.

CYSE 270 Assignment #9 Shell ScriptingWorking with Directory

References

Rothwell, W. (2019, p.268). Linux essentials for cybersecurity lab manual. Pearson It Certification.

JOHN WILSON
3/21/22

CYSE 270 Assignment #9 Shell ScriptingWorking with Directory