Old Dominion University

CYSE 270 Linux System for Cybersecurity

Assignment #9 Shell Scripting 2

Alysia Beckles

01127384

**Below is the snippet of a sample lab report.**

Task A

1. Write a shell script like below, that performs the following task:
* Reads the name of the directory.
* Check whether the given input is a directory or regular file.
* If the input is a directory and it exists, then display the message “Directory exists. Do not create”.
* If the input is a regular file, then display the message “Directory exists. Do not create”.
* If the given input name in step-1 doesn’t exist, then create the new directory with the given name in step-1.





**Your explanation goes here. For example:**
Above I utilized the if and else if conditional statements in order to create the outputs specified for this lab since there were multiple conditions needed to execute the shell script. To begin with I used shebang to execute the code as a BASH script. In order to make sure that the name of the directory is read I typed the command shell echo “Enter the directory” to be the output while the read dir\_name shell command is meant to gather user input of the specified directory name. In order to check whether the given input is a directory, I utilized the shell command if [ -n “$dir\_name” ] && [ -d “$dir\_name” ] for the directory input; -n is meant to evaluate and examine the command shell next to it while -d is true when the file is a directory. With the “and” operation both inputs would have to be true resulting to the display message that “Directory exists. Do not create.”, which is typed in the shell script as then echo “Directory exists. Do not create.” In order to check whether the given input is a regular file, I typed the shell command elif [ -f “$dir\_name” ]; then echo “This is a regular file! Do not create!”. I utilized elif because it enables me to utilize multiple expressions and if the if shell command is not true, the input will go to the elif shell command in order to see if it is true. The -f means that it is true if the file is a regular file. If the condition is true, the output will be “This is a regular file!Do not create!”. If both conditions end up being false the output will be “Directory does not exist, creating test directory now” and a new directory will be created. In order for the message to be displayed I typed the command else echo “Directory does not exist, creating test directory now” in the shell script. Also, I had to utilize the shell command mkdir -p “$dir\_name” to make a directory if the directory did not exist. The mkdir -p will make a parent directory if the directory does not exist. Next, I executed the shell script by typing ./directory.sh on the terminal. When I typed /home under “Enter the directory”, the output was “Directory exists. Do not create.” When I typed /etc/passwd under “Enter the directory”, the output was “This is a regular file!Do not create!”. Finally, when I typed test\_dir1 under “Enter the directory”, the output “Directory does not exist, creating test directory now”. Then, I typed the ls -l command to see that test\_dir1 was created as a directory.