

Systems Scripting

Semester 2, 2022

Assignment 1: Bash scripting

Submission Deadline: Wednesday 16th March 2022 Midnight

For each task write a short report describing your solution concept and how you arrived at it. This should be done on the top of each solution script file as comments. Failure to do so attracts penalty.

Task 1

Write a bash script that searches for patterns in files located in a folder. The script should be called with two input parameter arguments. Make sure that the arguments are provided before proceeding. The first parameter should be a path to a folder and the second parameter should be a string pattern.

The script should search the provided folder and print out the following details for only files identified (that is no sub-folder should be considered):

- Name of the file.
- Date and time of file creation.
- Size of the file in bytes.
- How many times the input string pattern (second parameter) appeared in the file (case insensitive).

Use an array structure to store the file names for those that contain the input string pattern (second parameter) at least twice.

An until loop should be used to iterate through the above array and print out to the terminal all the file names as well as write them into a file named report.txt. Use comments to properly document your script.

[32 marks]

Task 2

Write an interactive bash script that implements a set of menus for creating and writing contents into a file, outputting to the terminal the content of a file, change file permission and to terminate the script. The write, output and permission change operations should be implemented using functions.

When the user selects the write option, the script should demand for a file name, create it if it does not exist, and continuously demand for inputs and write them to the file until the user enters the word “**stop**” then the script should finish writing and return back to the menu options.

When the output option is selected, the script should demand for the name of the file to be printed and output its content. The script should ensure that the file exist and not empty before outputting all of its content to the screen. Then return back to the menu options.

When the permission option is selected, the script should demand for a name of the file, check if the file exist and assign execution permission to the file if not already assigned. In case execution permission is already assigned, simply report it. The terminate option should end the script with a goodbye message. Properly comment your code.

[32 marks]

Task 3

Write a bash script that automates the creation and deletion of user accounts. The script should accept as input argument, a file containing a list of user names to be created on the system. Enforce that the user provides this input file when running the script.

The script should check if the usernames already exist on the system before creating the accounts. If a user account exist, the script should notify the user and skip that user name to the next one. Make sure to create a home directory as well. When the input list has been exhausted and all the user account created, output the content of “/etc/passwd” file and “/home” directory to the terminal for verification.

In a next step, the script should ask if user wants to delete the newly created accounts? If yes, the script should delete the accounts including their home directories and output again the content of “/etc/passwd” file and “/home” directory for verification. If no, the script should terminate with appropriate message.

Use functions to implement the account creation and deletion operations. The functions should in each case accept one parameter. **This script should only be tested/executed with root user privileges. Ensure its enforcement.** Properly comment your code.

[36 marks]

Submission Instruction:

Each task should be solved with a different script file. Put the solution scripts into a folder and name the folder as follows: **firstname-surname**. Create a Zip archive from the folder. It should have the name: **firstname-surname.zip** E.g., Vincent-Emeakaroha.zip. Upload this archive to Canvas.

Submission Deadline: Wednesday 16th March 2022 Midnight