Challenge 6: Bash Scripting

Writing Bash scripts is a powerful way to interact with our Linux system. Bash scripts can be used any time you have a repeated set of commands that you use on a regular basis. In this example, we are going to write a bash script that will encrypt or decrypt a given file for us using the gpg command.

Step By Step Guide:

- Open up a terminal and navigate to your Scripts directory that we created in our previous videos.
- Create a new file called ed (for encrypt/decrypt).
- Open this file to make edits.

- Indicate what interpreter should be used for this script (we want to use bash).
- Put a comment describing the use of the script.

- Display a welcome message to the screen stating that the program will encrypt or decrypt a given file.
- Prompt the user to enter a filename to work with.
- Read the filename.

- Set a variable filepath which is the absolute path of the given filename. This will be important such that we can still run the script correctly if we are not in the same folder as the script.

- Prompt the user to enter if they would like to encrypt or decrypt a file.
- Read their choice.
- Echo a blank line for formatting purposes.

- If the given file is a regular file and exists:
    - If the user wants to encrypt:
        - Use the gpg command to encrypt the given file.
            - You will want to view the manual for gpg.
            - Make sure you do not cache the password.
        - Echo that the file has been encrypted.
        - Remove the original file.
    - Elif the user wants to decrypt:
        - Use the gpg command to decrypt the given file.
            - You will want to view the manual for gpg.
    - Else:
        - Echo that the user's choice was not a viable option.
- Else:
    - Echo that the file is not a regular file.
    - For all the files in the current directory:

        - If the current file is a regular file:
            - Echo a message that the file could be used with this script.

- Echo a blank line for formatting purposes.
- Echo the current encrypted files in the given directory.

- Close the file and save all changes.
- Set the permissions of ed such that it can be executed.

- Create 3 text files and populate each of them with text: file1, file2, and file3
- Create 2 directories: test1 and test2

- Run your script to encrypt the directory test1.
- Run your script to encrypt the file file1.
- Try to view the contents of file1.

- Run your script to decrypt file1.
- Verify the original contents.

- If the script is working, move the script to /usr/local/bin and run the script in any directory to verify that it is still working.

- Remove the files created for this video: file1, file2, file3, test1, and test2.