

CSCI-130 Linux Fundamentals - Week 3 (September 9, 2025) Assignment Submission

Assignment Details

1. Working with Files and Directories

1. Create a few nested and branched directories in your home directory. Populate them with some files. Copy some files around, use the move command to rename files or move files between directories.
2. Try to delete a non-empty directory with the `rmdir` command. What do you see? What should you do to delete the directory?
3. Delete all other directories you created with the `rm -rf` command. Do `ls -l` from your home directory to verify. Make sure you provide the correct path to the `rm` command in this part.

Beware that you may delete something you did not want to.

2. Changing Permissions

1. Change permissions on some of the files to add 'x' (execute) and 'w' (write) permissions to you only, to the group, and to all. Do `ls -l` to verify changes. What do you see?
2. Do a web search and find out how to change permissions recursively.
3. Look at the screenshot below showing permissions for the `/tmp` directory. Do a web search and find out what is the purpose of the last bit (the 't'). What is the "sticky bit" and why is it needed?

How to Submit

- For problem 1: Submit a relevant history of your commands.
- For problems 2 and 3: Submit a PDF file with a summary of your research.

Assignment Submission Notes

Problem 1: Working with Files and Directories

History exported from terminal:

```
1  pwd
2  alias lsa
3  which ls
4  lsa
5  mkdir csci-130-assignment
6  mkdir csci-130-assignment/folder1
7  mkdir csci-130-assignment/folder1subfolder
8  mkdir csci-130-assignment/folder2
9  mkdir csci-130-assignment/folder2/folder2subfolder
10 touch csci-130-assignment/root{1..5}
11 touch csci-130-assignment/folder1/folder1_file{1..5}
12 lsa csci-130-assignment/
13 cp csci-130-assignment/folder1subfolder/ csci-130-
assignment/folder1/
14 cp -r csci-130-assignment/folder1subfolder/ csci-130-
```

```
assignment/folder1/
15  lsa csci-130-assignment/
16  lsa csci-130-assignment/folder1
17  touch csci-130-assignment/folder1/folder1subfolder/f1sub_file{1..5}
18  lsa csci-130-assignment/folder1/folder1subfolder/
19  touch csci-130-assignment/folder2/folder2_file{1..6}
20  ls -lat csci-130-assignment/
21  ls -la csci-130-assignment/
22  mkdir csci-130-assignment/deleteThisFolder
23  mkdir csci-130-assignment/deleteThisFolderWithFiles
24  touch csci-130-
assignment/deleteThisFolderWithFiles/recursiveDelete_file{1..100}
25  lsa csci-130-assignment/deleteThisFolder
26  lsa csci-130-assignment/deleteThisFolderWithFiles/
27  cp csci-130-assignment/folder1/* csci-130-
assignment/deleteThisFolderWithFiles/
28  lsa csci-130-assignment/deleteThisFolderWithFiles/
29  lsa csci-130-assignment/
30  mv csci-130-assignment/root5 csci-130-assignment/root0
31  lsa csci-130-assignment/
32  rmdir csci-130-assignment/deleteThisFolderWithFiles/
33  rm -rf csci-130-assignment/deleteThisFolderWithFiles/
34  lsa csci-130-assignment/
35  rmdir csci-130-assignment/deleteThisFolder/
36  lsa
37  lsa csci-130-assignment/
38  rm -rf csci-130-assignment/
39  lsa
40  ls -l
41  history > csci-130-assignment-lab1.txt
```

Problem 2: Changing Permissions

History exported from terminal:

```
1  touch permission-test-file
2  ls -l
3  echo 'Permissions are -rw-rw-r--'
4  chmod u+xw permission-test-file
5  ls -l
6  chmod +xw permission-test-file
7  ls -l
8  chmod o+xw permission-test-file
9  ls -l
10 echo 'Permissions are -rwxrwxrwx'
11 chmod -R 777 Public/
12 ls -l
13 history > csci-130-assignment-lab1-problem2.txt
```

When you change permissions recursively with the **-R** option, it applies the permission changes to all files and directories within the specified directory, including subdirectories and their contents. This is useful when you want to ensure that all files and directories within a certain path have the same permissions.

Problem 3: Sticky Bit

The sticky bit is a special permission that can be set on directories to enhance security. When the sticky bit is set on a directory, only the owner of a file within that directory, the owner of the directory, or the root user can delete or rename the files. This is particularly useful in shared directories like **/tmp**, where multiple users have write access. In the case of the **/tmp** directory, the sticky bit prevents users from deleting or renaming files that they do not own, even though they have write permissions to the directory. This helps to prevent accidental or malicious deletion of files by other users. You can identify that the sticky bit is set on a directory by looking for a **t** at the end of the permission string. For example, in the permission string **drwxrwxrwt**, the **t** indicates that the sticky bit is set.

Here is an example of the permission string for the **/tmp** directory:

```
drwxrwxrwt 10 root root 4096 Sep  9 12:00 /tmp
```

In this example, the **t** at the end of the permission string indicates that the sticky bit is set on the **/tmp** directory.

Here is how to set the sticky bit on a directory:

```
$ chmod +t directory_name
$ ls -ld directory_name
drwxrwxrwt  2 user user 4096 Sep  9 12:30 directory_name
```

This command adds the sticky bit to the specified directory, enhancing its security by restricting file deletion and renaming permissions.