LONDON METROPOLITAN UNIVERSITY

informatics college • pokhara

Module Code & Module Title

Level 5 - CT5052NI

Assessment Weightage & Type

Individual Report Writing

Year and Semester

2023-24 Autumn

Student Name: Aman Bahadur Paudel

London Met ID: 23048978

College ID: NP04CP4A230136

Assignment Submission Date: 21/12/2024

Submitted To: Mr. Prashant Adhikari

Declaration: I understand that I am required to submit my assignment under the appropriate module page prior to the specified deadline, in order for it to be considered for marking. I acknowledge that any assignment submitted after the deadline will be deemed as a non-submission and will not be marked, resulting in a score of zero.

Table of Contents

Table of Figures

1. Introduction

Linux is a powerful open-source operating system that can be widely used in various domains, including system administration, cybersecurity, and software development. This lab report focuses on gaining hands-on experience with essential Linux commands. It provides a foundation for managing files, users, and system processes efficiently. After practicing these commands, one can navigate and operate a Linux-based system and enhance the ability to perform administrative tasks.

2. Objective

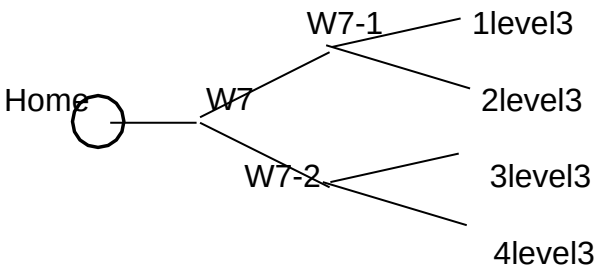
The main objective of this workshop is to be familiar with the Linux commands and the related utilities.

3. Required Tools

Linux or Linux based operating system. Here, I am using the Terminal Emulator from Kali Linux.

4. Practiced contents

1. Create the directory structure presented in the figure below. Use **mkdir** command and relative pathnames from your home directory. Try both: no option and **–p** option, for the command.



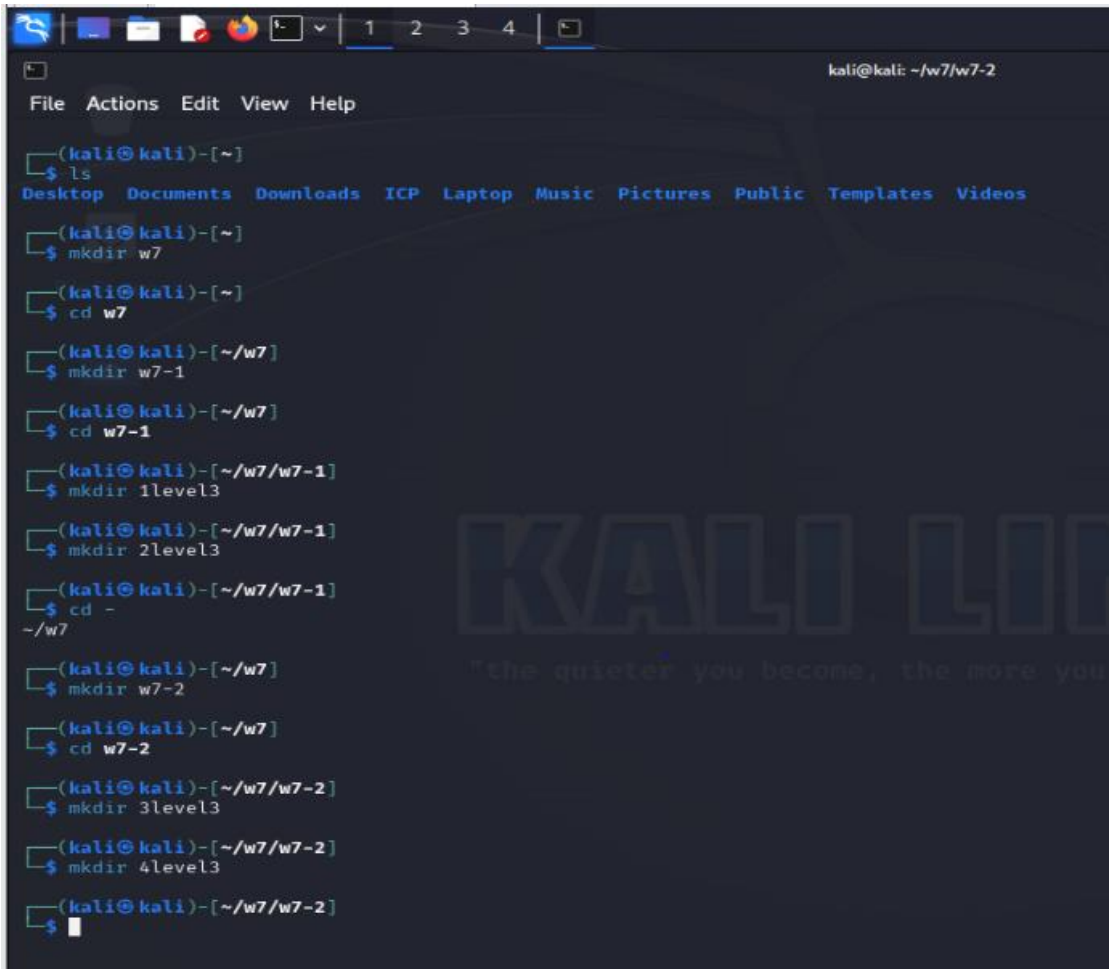


Figure 1: Creating directory without -p

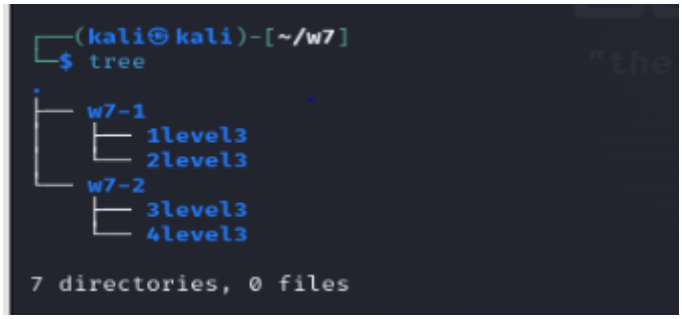


Figure 2: Created directory without -p

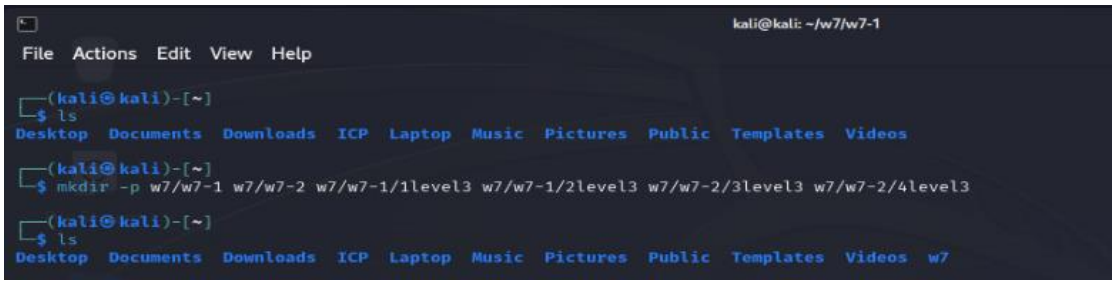


Figure 3: Creating directory by using -p

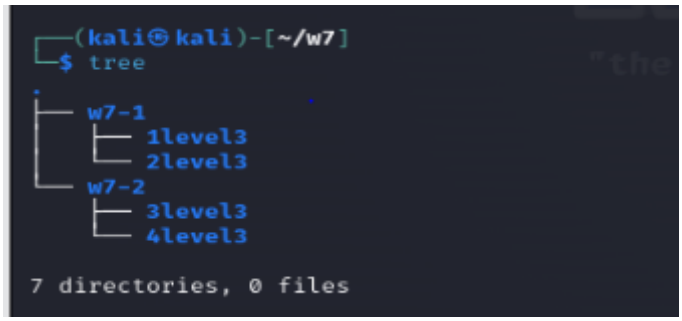


Figure 4: Created directory by using -p

2. Change to the **1level3** directory by one step using a relative pathname.

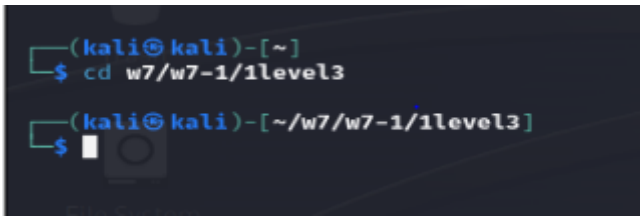


Figure 5: Changing directory using relative pathname

3. Practice in changing directories in your directory structure by one command using relative pathnames, e.g., from **1level3** to **2level3**, from **2level3** to **4level3**, from **4level3** to **W7**, etc. Use names of parent and child directories ('**.**' and '**..**') as well.

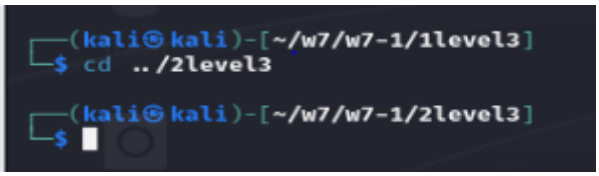


Figure 6: Directory changed from 1level3 to 2level3

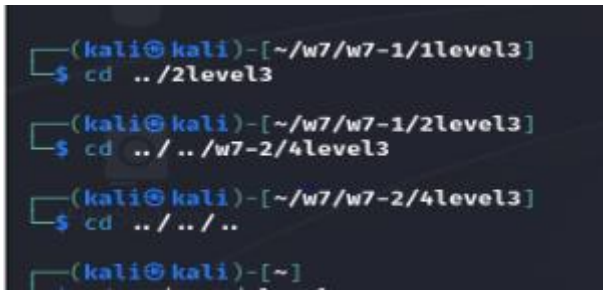


Figure 7: Practicing changing directory

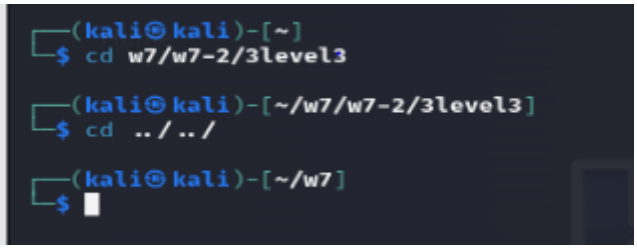


Figure 8: Directory changed from 3level3 to w7

4. Change to **1level3** and create a text file by any tool (e.g., by **cat** or **cal** like last tutorial).

Figure 9: Creating text file using cat command

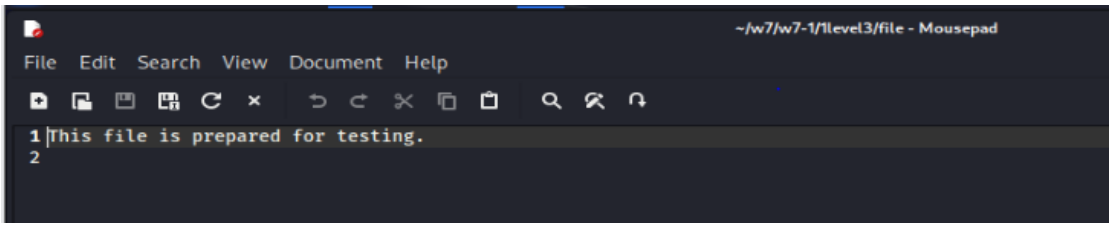


Figure 10: Created file

5. Copy this text file from **1level3** to **1level3** (with the name **file1**), **2level3**, and to **3level3** changing its name. Show that there are these files in corresponding directories.

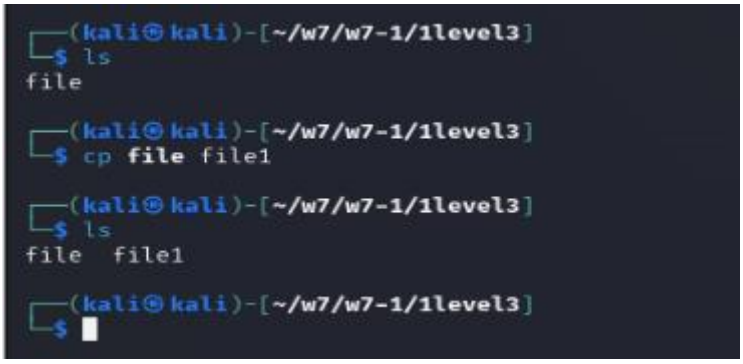


Figure 11: File copied to same directory as file1

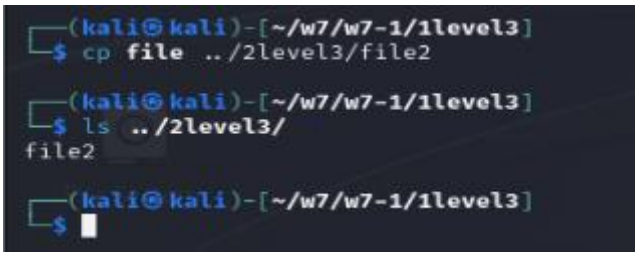


Figure 12: File copied to 2level3 as file2

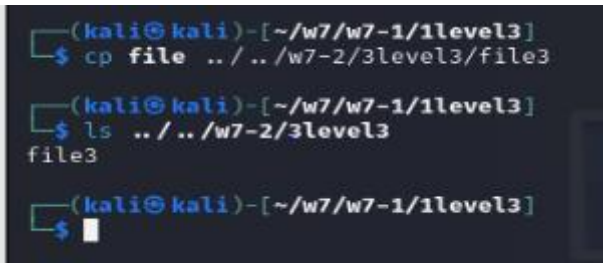


Figure 13: File copied to 3level3 as file3

6. Move this file to **4level3**. Show that there is this file in **4level3** and there is not in **1level3**.

Figure 14: File moved from 1level3 to 4level3

7. Print the following texts each in one **echo** or **printf** command:

   Hello! I can do it
   5 > (20: 8) < (30 * 2)

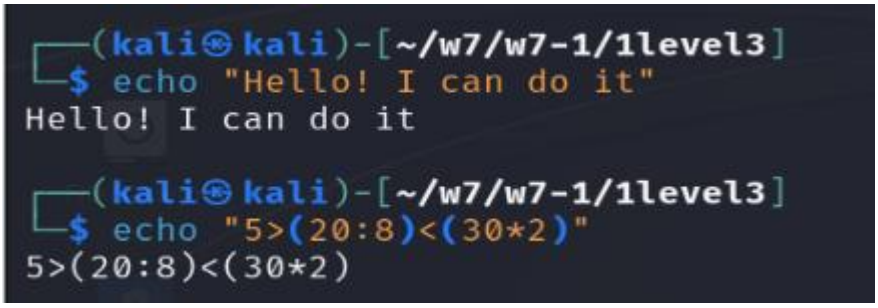


Figure 15: Printing the text using echo command

8. Give the **ls** command (without options and with **a**, **d**, **g**, **l**, **R** options) in home directory, **w7**, **w7-1**, and **1level3** directories. Explain for yourself the results received.

Figure 16: Using ls command with and without a, d, g, l options in w7 directory

Figure 17: Using ls command with and without R option in w7 directory

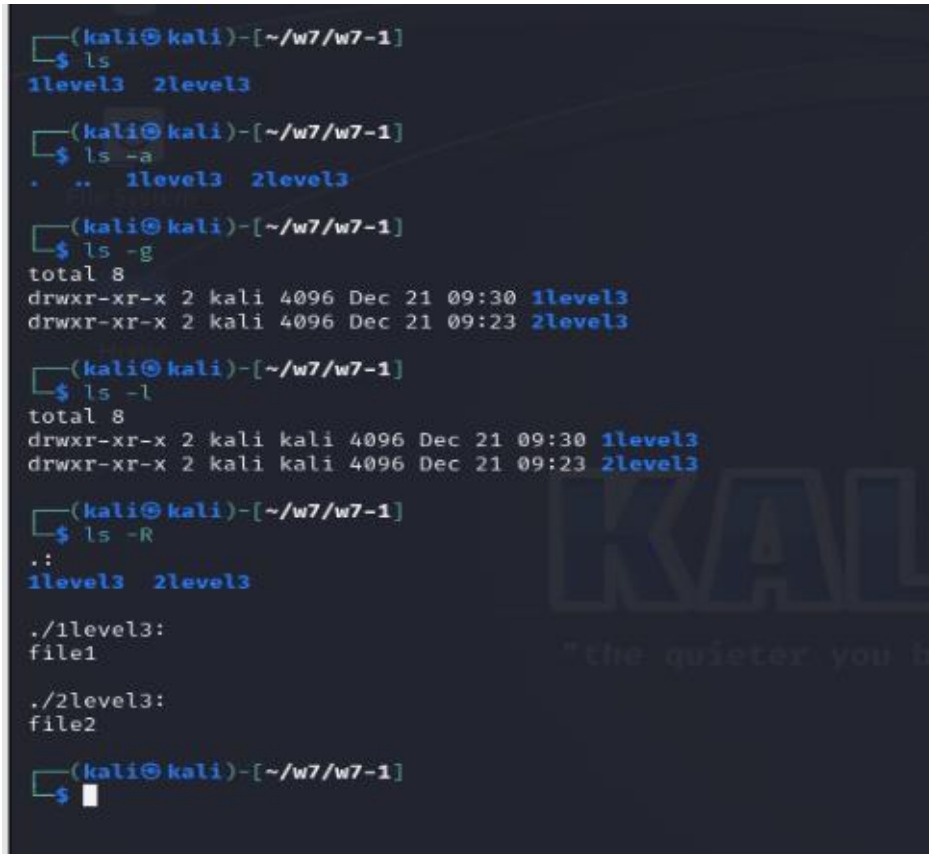


Figure 18: Using ls command with and without a, d, g, l, R options in w7-1 directory

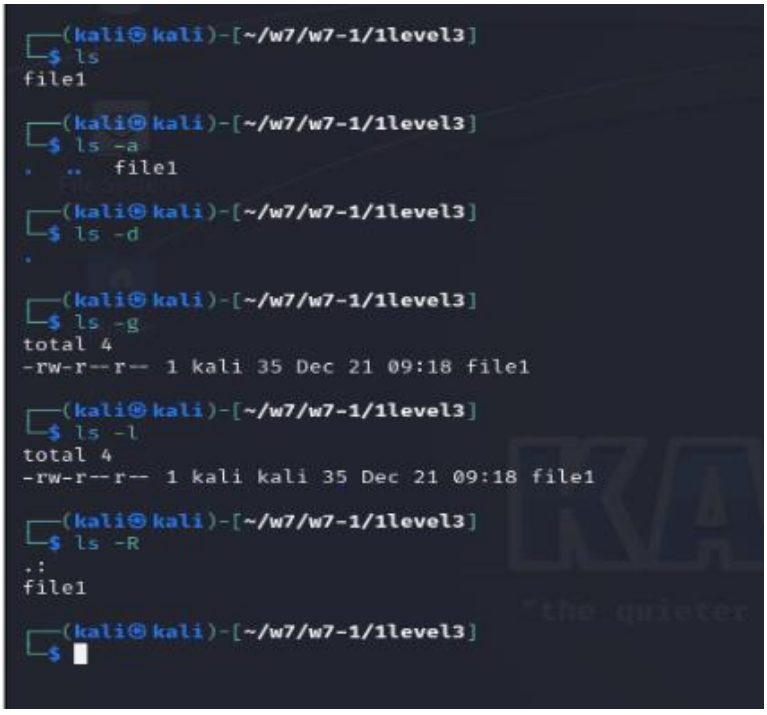


Figure 19: Using ls command with and without a, d, g, l, R options in 1level3 directory

9. Change to the **W7** directory. Remove the directory files **w7-2**, **3level3**, **4level3** and all ordinary files in them. Use the option **–i** of the **rm** and **rmdir** commands. Show that there are not these ordinary and directory files in your file structure.

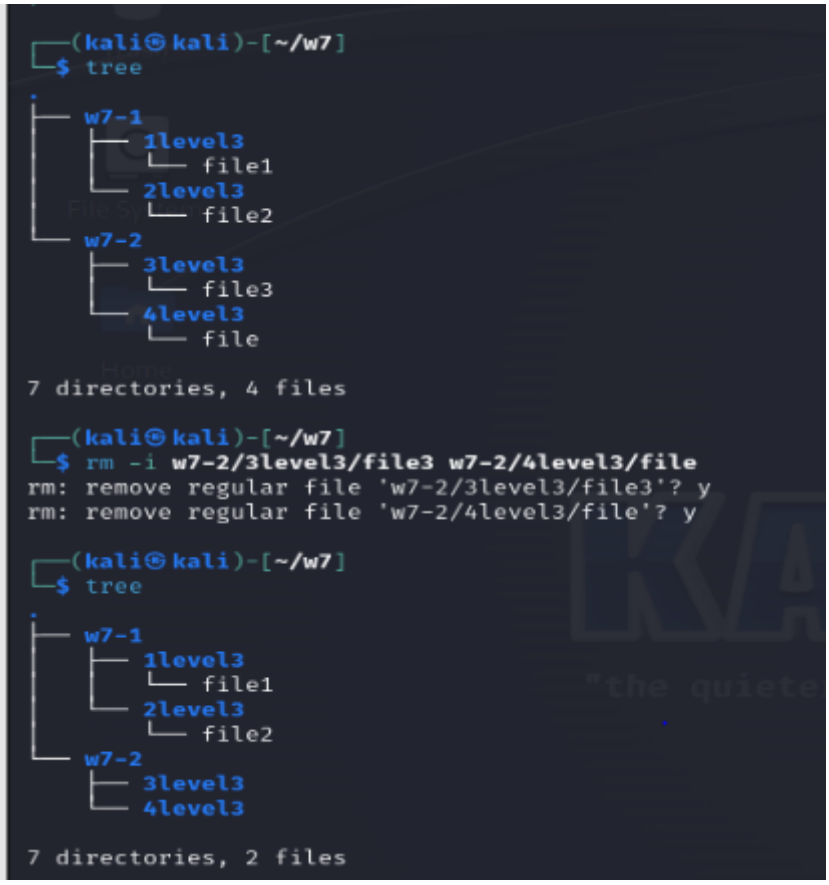


Figure 20: Removing the files: file and file3 from w7-2

Figure 21: Removing directory w7-2, 3level3, 4level3

10. Change to **w7-1**.
    a. Display access permissions for the file **file1** in **1level3**.

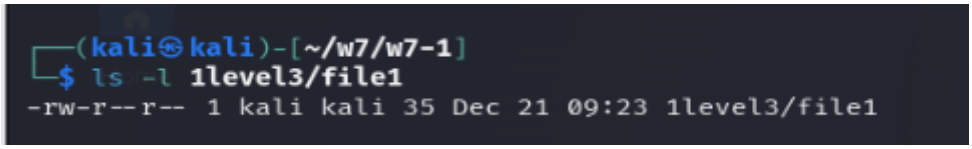


Figure 22: Access permission of file1

    b. Remove all access permissions for this file.

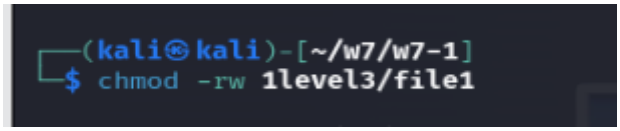


Figure 23: Removing access permission of file 1

    c. Display access permissions for this file.

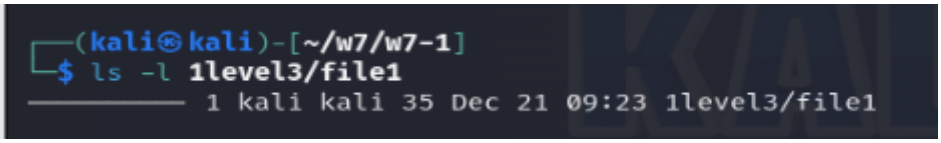


Figure 24: Access permission of file1

    d. Try to read this file using any utility (e.g., **cat**).

Figure 25: Trying to read file1

    e. Try to write into this file using any utility (e.g., **cat** with the sign **>>** – *append*).

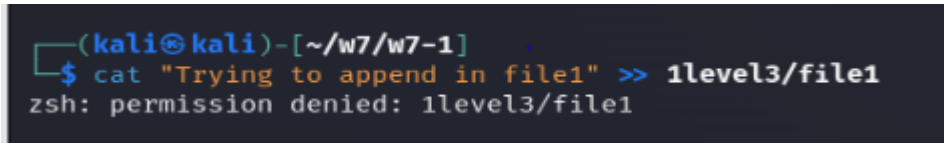


Figure 26: Trying t write into the file

f. Add read and write access permissions for yourself for this file.

```
┌──(kali㉿kali)-[~/w7/w7-1]
└─$ chmod 600 1level3/file1
```

Figure 27: Adding read and write access for self

g. Display access permissions for this file.

```
┌──(kali㉿kali)-[~/w7/w7-1]
└─$ ls -l 1level3/file1
-rw------- 1 kali kali 35 Dec 21 09:23 1level3/file1
```

Figure 28: Access permission of file1

h. Try to read this file using any utility.

```
┌──(kali㉿kali)-[~/w7/w7-1]
└─$ cat 1level3/file1
This file is prepared for testing.
```

Figure 29: Trying to read file1 again after permission granted

i. Try to write into this file using any utility.

```
┌──(kali㉿kali)-[~/w7/w7-1]
└─$ echo "Additional content adding after giving read and write permission to self" > 1level3/file1

┌──(kali㉿kali)-[~/w7/w7-1]
└─$ cat 1level3/file1
Additional content adding after giving read and write permission to self
```

Figure 30: Trying to write into file again after permission granted

11. (Now,)

a. Display access permissions for **1level3**.

```
┌──(kali㉿kali)-[~/w7/w7-1]
└─$ ls -l 1level3
total 4
-rw------- 1 kali kali 73 Dec 21 12:09 file1
```

Figure 31: Access permission of directory 1level3

b. Remove all access permissions for the **1level3** directory.

Figure 32: Removing access permission

c. Display access permissions for **1level3**.

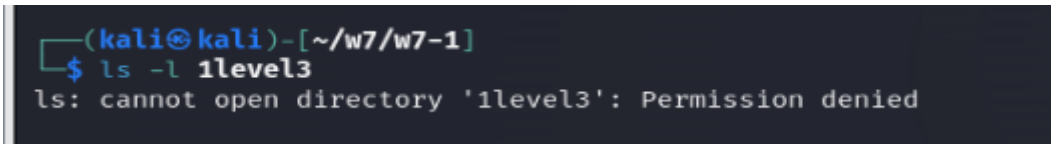


Figure 33: Access permission of directory 1level3

d. Try to read a file from **1level3** using any utility.

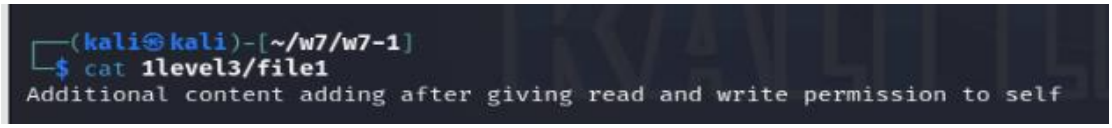


Figure 34: Trying to read file from directory 1level3

e. Try to put a file into **1level3** using any utility.

Figure 35: Trying to put file into directory 1level3

f. Try to search in **1level3** using any command (e.g., the **ls** command).

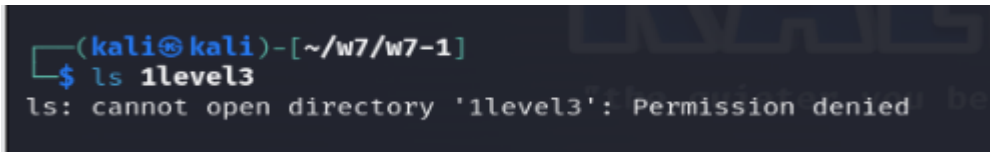


Figure 36: Searching file with ls command

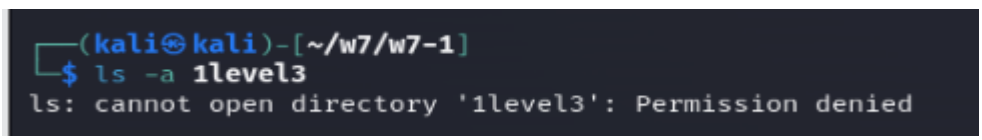


Figure 37: Searching file with ls -a command

g. Add read, write, and execute access permissions for yourself for the **1level3** directory.

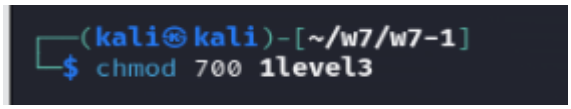


Figure 38: Adding access permission for self

h. Display access permissions for **1level3**.

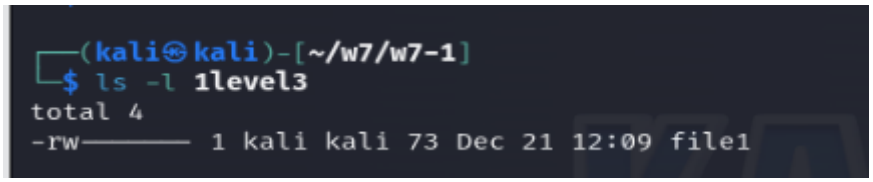


Figure 39: Access permission of directory 1level3

i. Try to read a file from **1level3** using any utility.

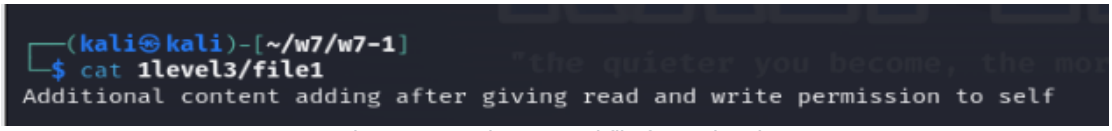


Figure 40: Trying to read file from 1level3

j. Try to put a file into **1level3** using any utility.

Figure 41: Trying to put file into 1level3 with cat command

k. Try to search in **1level3** using any command (e.g., the **ls** command).

Figure 42: Listing files of directory 1level3

5. Conclusion

This document has the detailed practical steps and theoretical knowledge required to manage directories and files in a Linux environment effectively. From basic commands like creating and navigating directories to more advanced operations such as modifying file permissions, each section provides essential skills.

By practicing these operations, we can gain a deeper understanding of Linux file systems, improve workflow efficiency, and strengthen their command-line proficiency. Whether creating structured directories, manipulating file permissions, or executing commands with various options, the tasks equip uss with the confidence to navigate real-world scenarios. Through these practices, we also can enhance security and accessibility in a multi-user environment.