

A penguin with yellow feet and blue text

Description automatically generated

~ Abhishek Adhikari

**About**

This project is completed as part of Task 2 for Linux Fundamentals, covering a sequence of commands to create users, manage groups, and set file permissions on a Linux system.

Submitted to:

Rikesh Maharjan

**2024**

[Linux User and Group Management Documentation 2](#_Toc174118693)

[1. Creating a New User and Adding to the `sudo` Group 2](#_Toc174118694)

[1.1 Objective 2](#_Toc174118695)

[1.2 Commands and Execution 2](#_Toc174118696)

[1.2.1 Create the User: 2](#_Toc174118697)

[1.2.2 Add the User to the `sudo` Group: 2](#_Toc174118698)

[1.3 Verification 3](#_Toc174118699)

[2. Creating a Group and Adding Multiple Users 4](#_Toc174118700)

[2.1 Objective 4](#_Toc174118701)

[2.2 Commands and Execution 4](#_Toc174118702)

[2.2.1 Create the group: 4](#_Toc174118703)

[2.2.2 Create New Users and Assign Passwords: 4](#_Toc174118704)

[2.2.3 Add Users to the `batch37` Group: 5](#_Toc174118705)

[2.3 Verification 6](#_Toc174118706)

[3. File Management and Permission Settings 6](#_Toc174118707)

[3.1 Objective 6](#_Toc174118708)

[3.2 Commands and Execution 7](#_Toc174118709)

[3.2.1 Create Home Directory and Set Permissions: 7](#_Toc174118710)

[3.2.2 Create and Write to a File: 7](#_Toc174118711)

[3.3.3 Change Ownership and Group Ownership: 8](#_Toc174118712)

[3.3.4 Modify File Permissions: 8](#_Toc174118713)

[3.3 Verification 9](#_Toc174118714)

[3.3.1 Verify Access for Group Members: 9](#_Toc174118715)

[3.3.2 Verify No Access for Others: 9](#_Toc174118716)

[4. Summary and Conclusion 10](#_Toc174118717)

[4.1 Summary 10](#_Toc174118718)

[4.2 Conclusion 10](#_Toc174118719)

[5. References 11](#_Toc174118720)

# Linux User and Group Management Documentation

## Creating a New User and Adding to the `sudo` Group

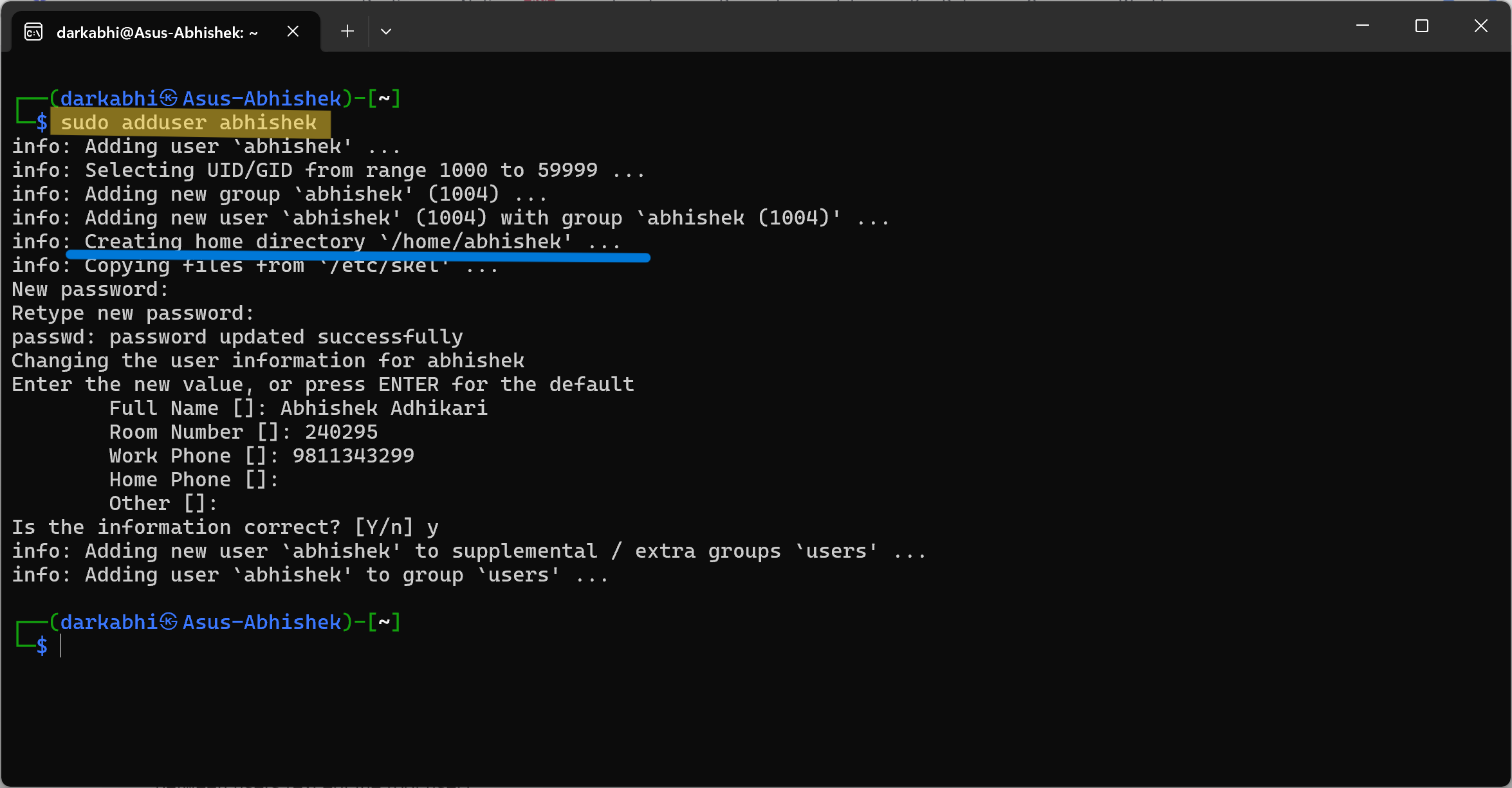
### 1.1 Objective

To create a new user with your name (e.g., abhishek) and add this user to the sudo group, enabling administrative privileges.

### 1.2 Commands and Execution

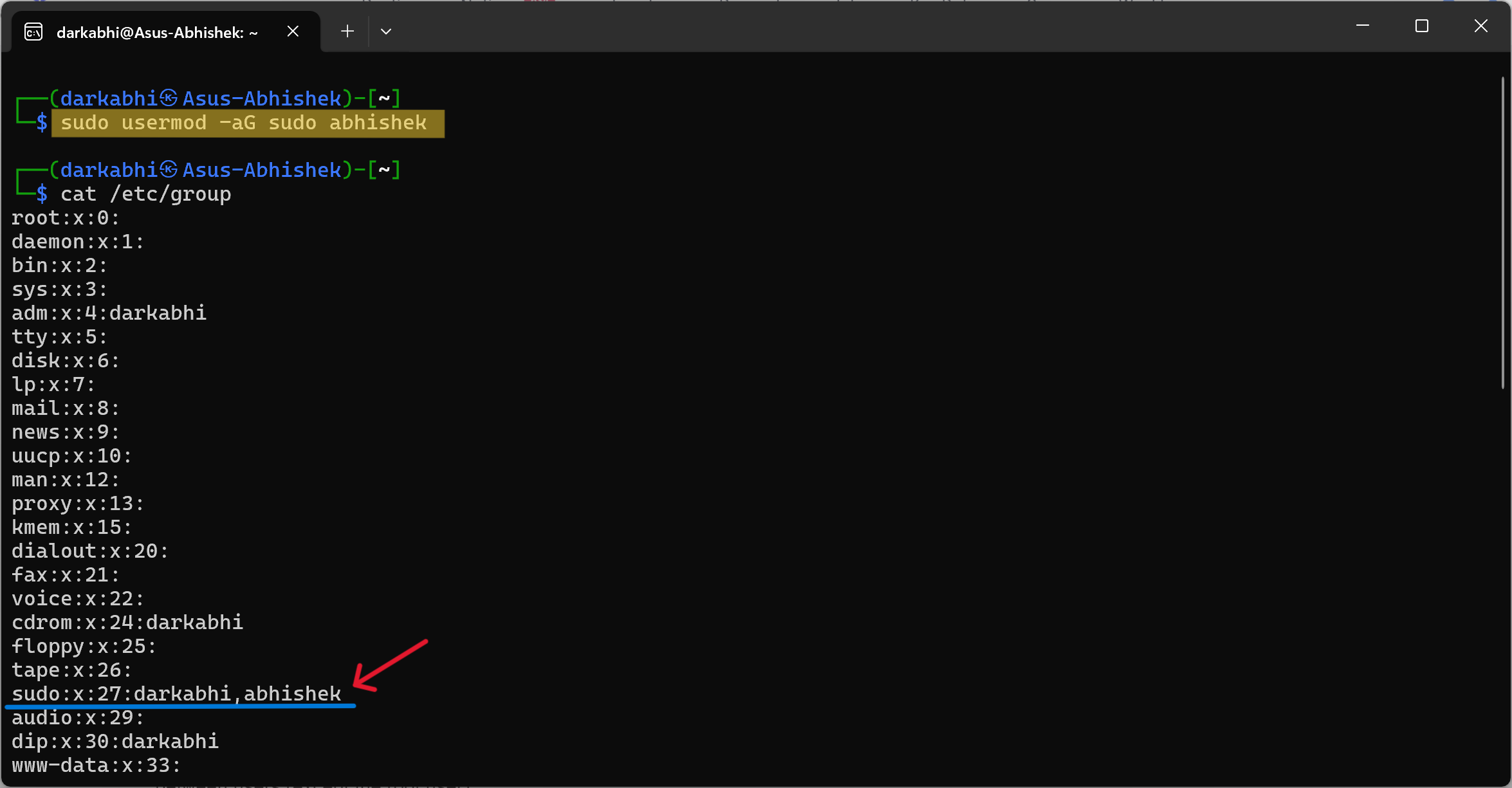
#### Create the User:

* Use the ` USERADD ` command to create a new user named “abhishek”.
* Command:
* sudo ` USERADD ` -m -s /bin/bash abhishek
* Explanation:
* `- m `: Creates a home directory for the user.
* `-s /bin/bash`: Sets the default shell to bash.
* Visual: The terminal screen shows the command.



#### Add the User to the `sudo` Group:

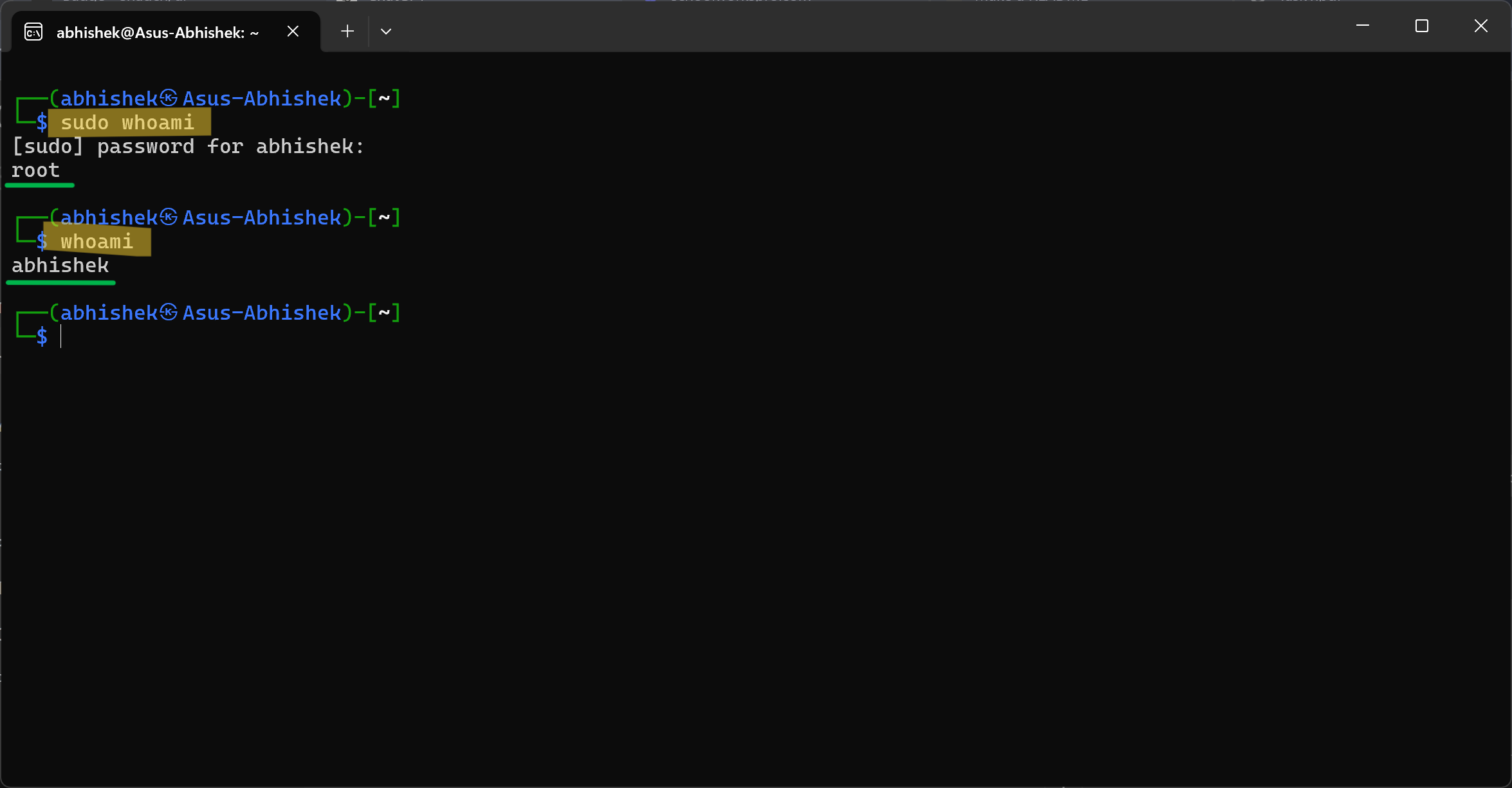
* Add the user “abhishek” to the sudo group for administrative privileges.
* Command:
* sudo ` USERMOD ` -aG sudo abhishek
* Explanation:
* `-aG`: Adds the user to the specified group without removing them from other groups.
* Visual: The terminal screen shows the command.



### 1.3 Verification

Verify that the user has been added and has sudo privileges by switching to the user and running a sudo command.

* Commands:
* ` SU ` abhishek
* ` SUDO ` whoami ……. (i)
* whoami ……. (ii)
* Expected Output: (i) ` root ` (ii) ` abhishek `
* Visual: The terminal screen shows the command.



## Creating a Group and Adding Multiple Users

### 2.1 Objective

Create a new group named “batch37” and add three new users (std1, std2, std3) to this group.

### 2.2 Commands and Execution

#### Create the group:

* Use the ` GROUPADD ` command to create a new group named “batch37”.
* Command:
* sudo ` GROUPADD ` batch37
* Explanation:
* This creates a new group in the system.
* Visual: The terminal screen shows the command.



#### Create New Users and Assign Passwords:

* Create users ` STD1 `, ` STD2 ` & ` STD3 ` and set their passwords.
* Commands:
* sudo ` USERADD ` std1
* sudo ` PASSWD ` std1
* sudo ` USERADD ` std2
* sudo ` PASSWD ` std2
* sudo ` USERADD ` std3
* sudo ` PASSWD ` std3
* Explanation:
* Similar to the steps taken to create the user abhishek, these commands create std1, std2 and std3 and set their passwords.
* Visual: The terminal screen shows the commands.

Create and set password for ` STD1 `.



Create and set password for ` STD2 `.



Create and set password for ` STD3 `.



#### Add Users to the `batch37` Group:

* Add the newly created users to the ` BATCH37 ` group.
* Commands:
* sudo ` USERMOD ` -aG batch37 std1
* sudo ` USERMOD ` -aG batch37 std2
* sudo ` USERMOD ` -aG batch37 std3
* Explanation:
* This command appends each user to the batch37 group without removing them from any other groups.
* Visual: The terminal screen shows the commands.





### 2.3 Verification

Verify the users are part of the batch37 group.

* Commands:
* ` GROUPS ` std1
* ` GROUPS ` std2
* ` GROUPS ` std3
* Expected Output: Each command should list ` BATCH37 ` among the groups.
* Visual: The terminal screen shows the command.



## File Management and Permission Settings

### 3.1 Objective

Create a file in `std1's` home directory, manage its ownership and permissions, and verify these settings.

### 3.2 Commands and Execution

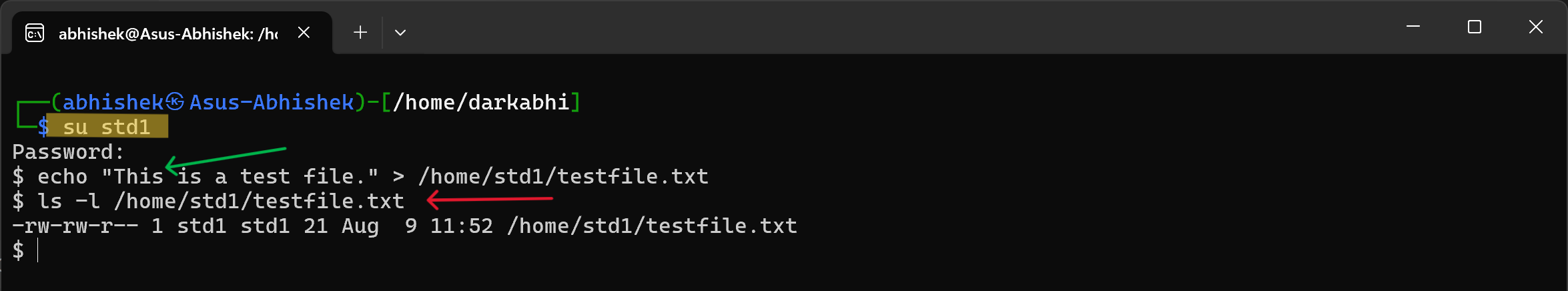
#### Create Home Directory and Set Permissions:

* Since the home directory for ` STD1 ` was not automatically created during user creation, it needs to be created manually, followed by setting ownership and permissions.
* Commands:
* sudo ` MKDIR ` /home/std1
* sudo ` CHOWN ` std1:std1 /home/std1
* sudo ` CHMOD ` 755 /home/std1
* Explanation:
* The `mkdir` command is used to manually create the home directory for the user std1.
* The `chown` command changes the ownership of the /home/std1 directory to user std1 and group std1.
* The `chmod` 755 command sets the permissions so that the owner has full access (read, write, execute), the group can read and execute, and others can also read and execute.
* Visual: The terminal screen shows the command.



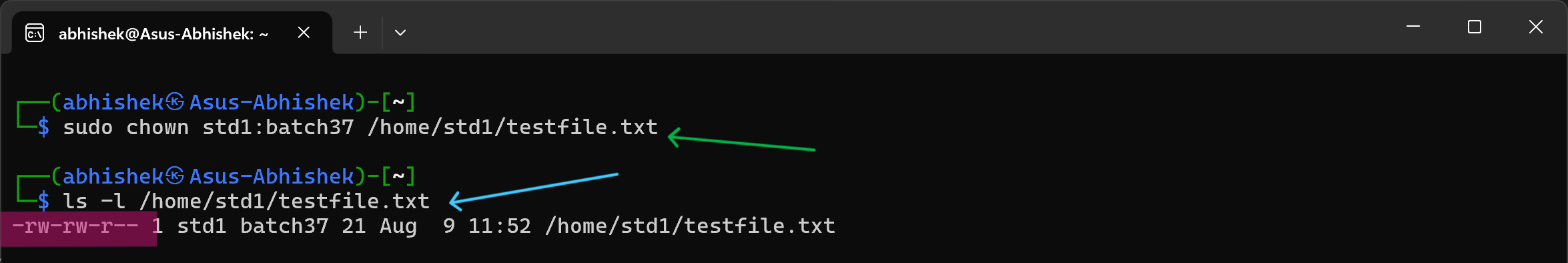
#### Create and Write to a File:

* Login as ` STD1 ` and create a new file in the home directory.
* Commands:
* su ` STD1 `
* ` ECHO ` "This is a test file." > /home/std1/testfile.txt
* ` LS ` -l /home/std1/testfile.txt
* Explanation:
* The echo command writes the string to the file testfile.txt.
* The ls command long lists the created file testfile.txt
* Visual: The terminal screen shows the command.



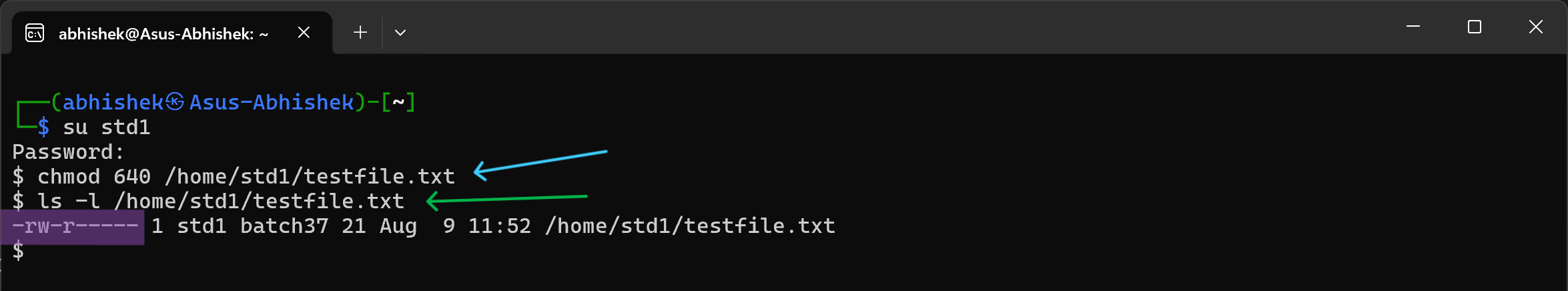
#### Change Ownership and Group Ownership:

* Change the ownership of the file to ` STD1 ` and the group ownership to ` BATCH37 `.
* Commands:
* sudo ` CHOWN ` std1:batch37 /home/std1/testfile.txt
* ` LS ` -l /home/std1/testfile.txt
* Explanation:
* Chown changes the file's ownership to std1 and group ownership to batch37.
* The ls command long lists the created file testfile.txt
* Visual: The terminal screen shows the command.



#### Modify File Permissions:

* Change the file permissions so that the owner can read and write, the group can only read, and others have no access.
* Commands:
* su ` STD1 `
* ` CHMOD ` 640 /home/std1/testfile.txt
* ` LS ` -l /home/std1/testfile.txt
* Explanation:
* Chmod 640 sets the permissions such that the owner can read/write, the group can read, and others have no permissions. [ **4** = Read (r), **2** = Write (w), **1** = Execute (x), **0** = No permissions (—) ]
* Visual: The terminal screen shows the command.

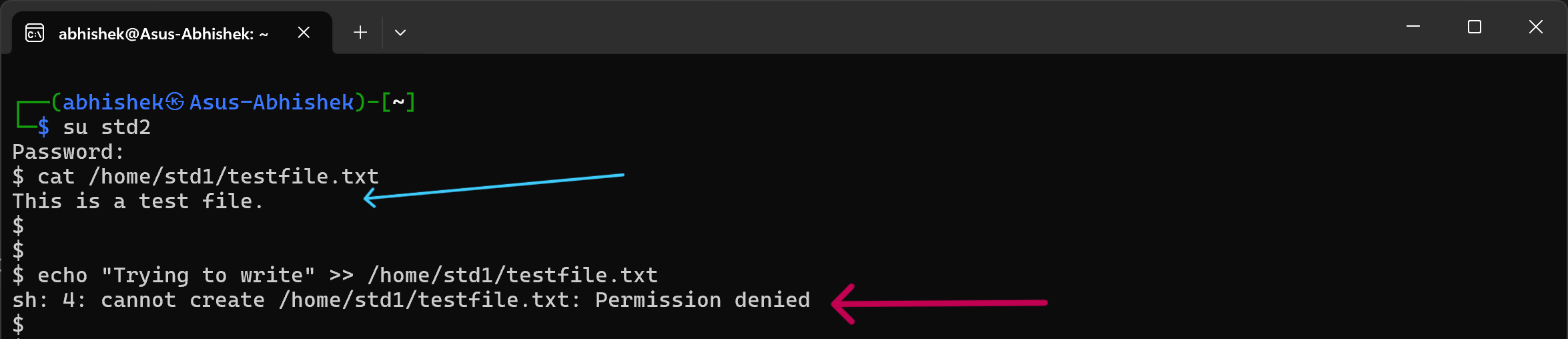


### 3.3 Verification

#### Verify Access for Group Members:

Verify Permissions by Switching Users.

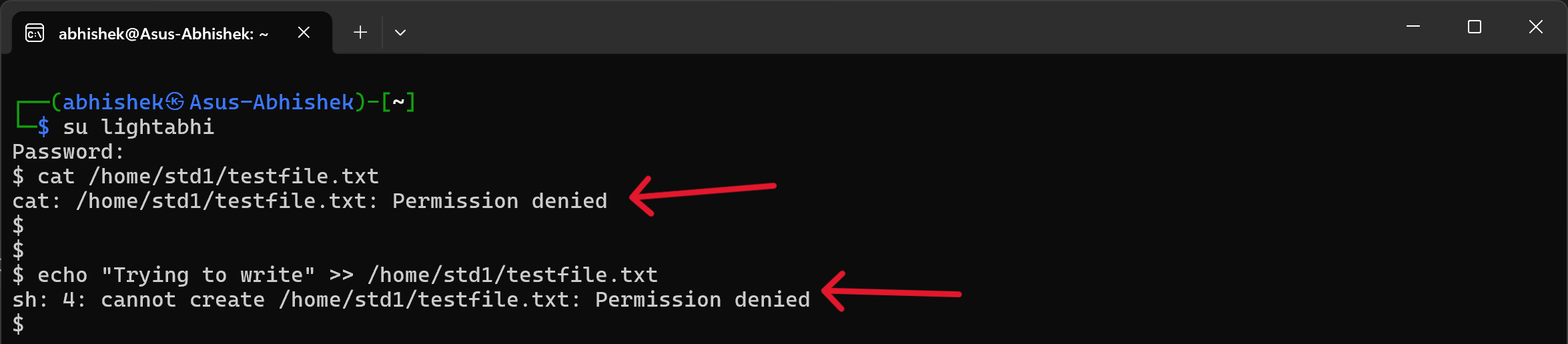
* Commands:
* ` SU ` std2
* ` CAT ` /home/std1/testfile.txt
* ` ECHO ` “Trying to write” > /home/std1/testfile.txt
* Expected Output: Both ` STD2 ` & ` STD3 ` should be able to read the file but should not be able to write to it. Members in groups should have only read permission.
* Visual: The terminal screen shows the command.



#### Verify No Access for Others:

Switch to another user not in the ` BATCH37` group and check access.

* Commands:
* ` SU ` lightabhi
* ` CAT ` /home/std1/testfile.txt
* ` ECHO ` “Trying to write” > /home/std1/testfile.txt
* Expected Output: The user should not be able to read or write to the file.
* Visual: The terminal screen shows the command.



## Summary and Conclusion

### 4.1 Summary

In this project, we successfully completed a series of tasks centered around user and group management in a Linux environment. The steps included:

* **Creating a New User and Assigning Sudo Privileges:** We created a user named abhishek, set a password, and added the user to the sudo group, granting administrative privileges.
* **Group Creation and User Assignment:** We created a new group named batch37 and added three users (std1, std2, and std3) to this group.
* **File Management and Permissions:** We manually created a home directory for the std1 user, set the appropriate ownership and permissions, and created a test file within this directory. We adjusted the file permissions to ensure that std1 had full access, while group members had read-only access, and other users had no access.
* **Verification of Permissions:** Finally, we verified the file permissions by switching between users, ensuring that the access control settings were applied correctly.

### 4.2 Conclusion

User and group management are critical components of system administration in Linux, playing a pivotal role in maintaining both security and organization. By creating and managing users and groups, system administrators can control access to system resources, ensuring that users have the appropriate level of access to files, directories, and commands.

This project highlighted the importance of correctly configuring user directories, ownership, and permissions to prevent unauthorized access and potential security breaches. Additionally, the manual creation of user home directories and subsequent permission setting underscored the flexibility and power of Linux in handling various administrative tasks.

Throughout the process, some challenges were encountered, particularly with ensuring the correct permissions and ownership settings. However, these challenges were addressed by careful command execution and verification, emphasizing the need for precision in system administration.

## References

**Official Ubuntu Documentation: User Management**

* [Link: https://ubuntu.com/server/docs/administration-users](https://ubuntu.com/server/docs/administration-users)
* <https://manpages.ubuntu.com/manpages/focal/man1>
* <https://manpages.ubuntu.com/manpages/focal/man1/mkdir.1.html>

**Linux man Pages**

* Commands like useradd, usermod, groupadd, chmod, chown, and passwd were referenced through the built-in manual (man) pages in Linux.

**The Linux Command Line by William E. Shotts, Jr.**

* A comprehensive guide for understanding Linux commands and system administration tasks.