Experiment 3: Advanced Filed and Directory Operations

Experiment 3: Linux File Manipulation and System Manipulation

Name: Hrithvik Bhardwaj Roll No.: 590029169 Date: 2025-09-23

Aim:

- To practice Linux file manipulation commands like `touch`, `cp`, `mv`, `rm`, `cat`, `less`, `head`, `tail`.
- To explore file permissions and ownership with `ls -l`, `chmod`, `chown`, and `chgrp`.
- To search and filter files using `find` and `grep`.
- To understand archiving and compression with `tar`, `gzip`, and `gunzip`.
- To create and manage links (`ln`) for both hard and symbolic links.

Requirements

- A Linux machine with bash shell (Ubuntu/Fedora/other).
- User privileges to create, modify, and delete files and directories.
- Access to system utilities like `tar`, `gzip`, `grep`, and `find`.

Theory

Linux file management involves creating, copying, moving, removing, and viewing files. File permissions and ownership ensure secure access control. Searching and filtering tools like `grep` and `find` help locate information efficiently. Archiving with `tar` and compression with `gzip` reduce storage usage and simplify file transfer. Links (`ln`) allow multiple references to the same file data (hard links) or path references (symbolic links).

Procedure & Observations

Exercise 1: Creating and Managing Files

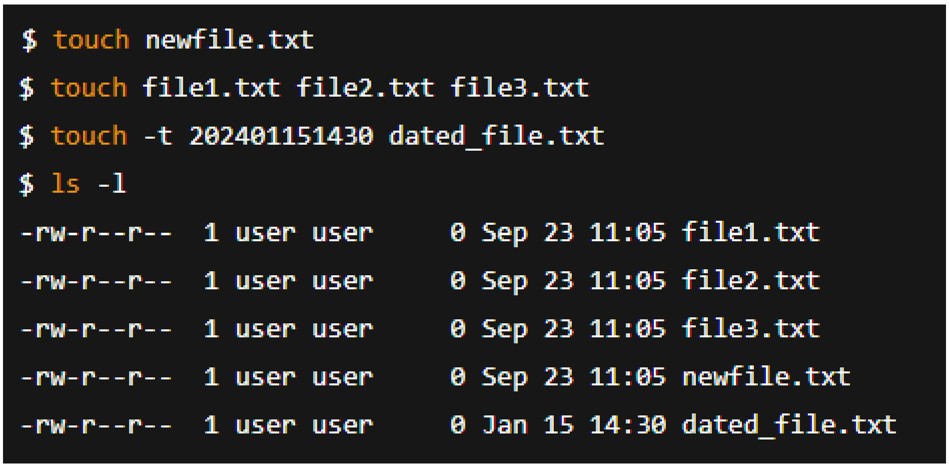
Task Statement:

Create files and manage timestamps using `touch`.

Command(s):

```
touch newfile.txt
touch file1.txt file2.txt file3.txt
touch -t 202401151430 dated_file.txt
```

Output:



```
$ touch newfile.txt
$ touch file1.txt file2.txt file3.txt
$ touch -t 202401151430 dated_file.txt
$ ls -l
-rw-r--r--  1 user user     0 Sep 23 11:05 file1.txt
-rw-r--r--  1 user user     0 Sep 23 11:05 file2.txt
-rw-r--r--  1 user user     0 Sep 23 11:05 file3.txt
-rw-r--r--  1 user user     0 Sep 23 11:05 newfile.txt
-rw-r--r--  1 user user     0 Jan 15 14:30 dated_file.txt
```

Exercise 2: Copying, Moving, and Deleting Files

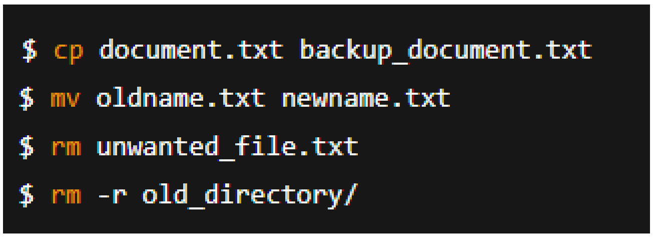
Task Statement:

Use `cp`, `mv`, and `rm` to copy, rename, move, and delete files and directories.

Command(s):

```
cp document.txt backup_document.txt
mv oldname.txt newname.txt
rm unwanted_file.txt
rm -r old_directory/
```

Output:



```
$ cp document.txt backup_document.txt
$ mv oldname.txt newname.txt
$ rm unwanted_file.txt
$ rm -r old_directory/
```

Exercise 3: Viewing File Contents

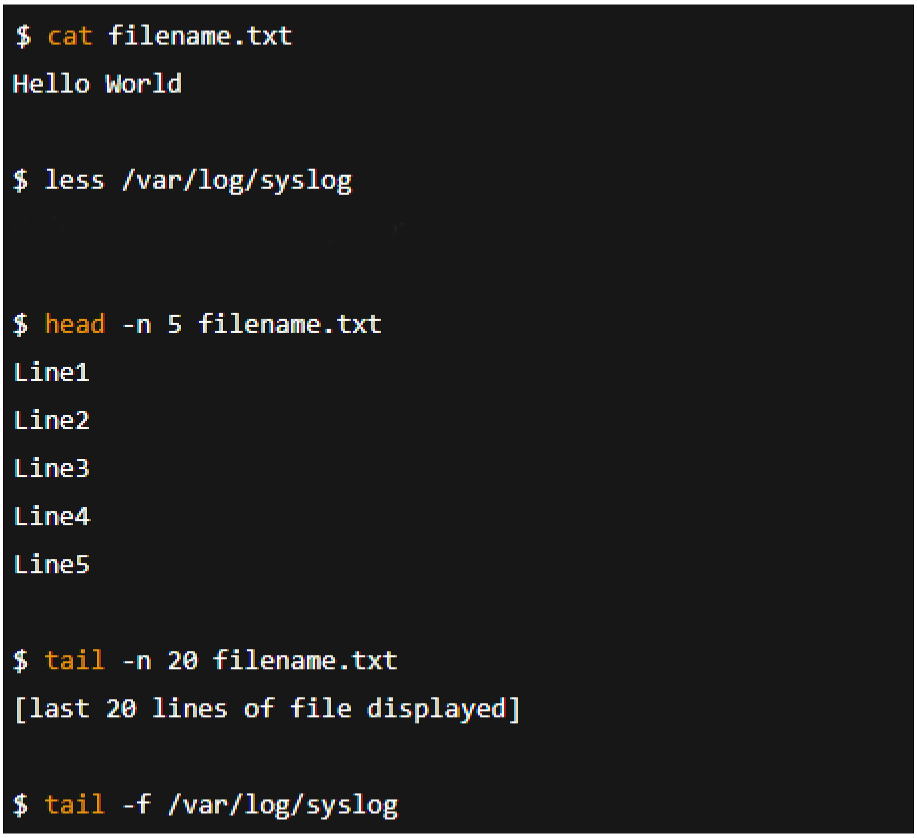
Task Statement:

Display file contents using `cat`, `less`, `head`, and `tail`.

Command(s):

```
cat filename.txt
less /var/log/syslog
head -n 5 filename.txt
tail -n 20 filename.txt
tail -f /var/log/syslog
```

Output:



```
$ cat filename.txt
Hello World

$ less /var/log/syslog



$ head -n 5 filename.txt
Line1
Line2
Line3
Line4
Line5

$ tail -n 20 filename.txt
[last 20 lines of file displayed]

$ tail -f /var/log/syslog
```

Exercise 4: File Permissions and Ownership

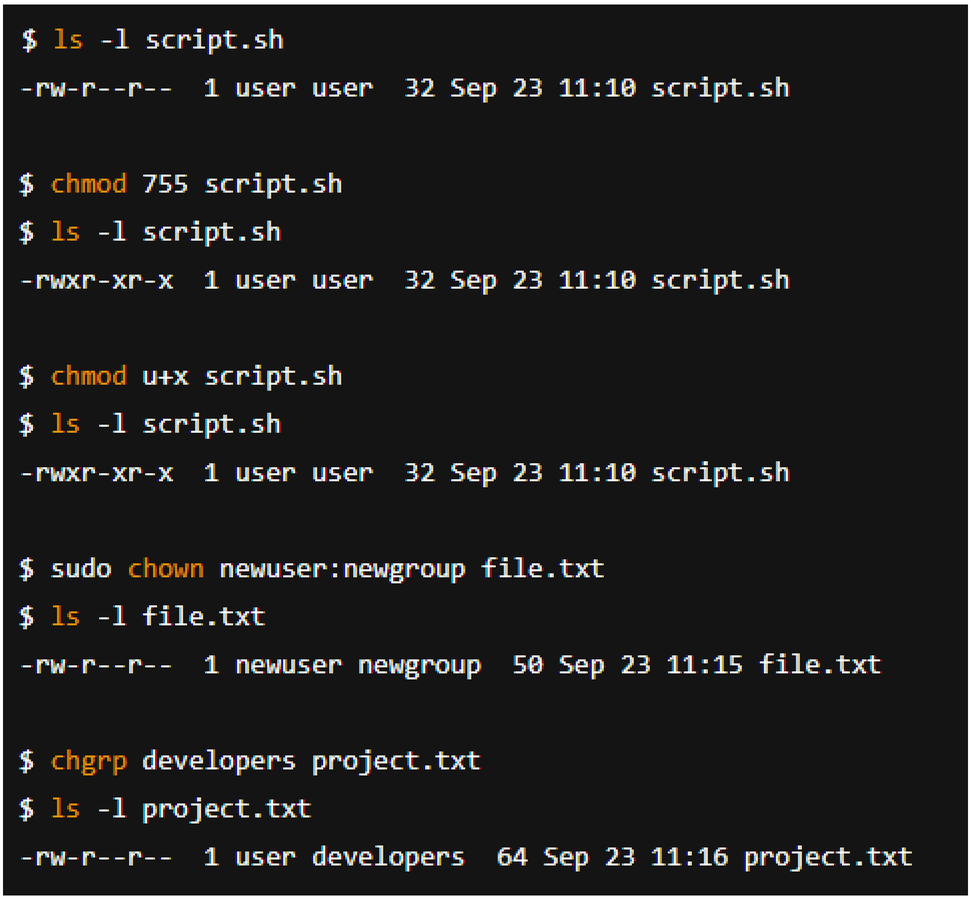
Task Statement:

Explore file permissions and ownership with `ls -l`, `chmod`, `chown`, and `chgrp`.

Command(s):

```
ls -l
chmod 755 script.sh
chmod u+x script.sh
sudo chown newuser:newgroup file.txt
chgrp developers project.txt
```

Output:



```
$ ls -l script.sh
-rw-r--r--  1 user user  32 Sep 23 11:10 script.sh

$ chmod 755 script.sh
$ ls -l script.sh
-rwxr-xr-x  1 user user  32 Sep 23 11:10 script.sh

$ chmod u+x script.sh
$ ls -l script.sh
-rwxr-xr-x  1 user user  32 Sep 23 11:10 script.sh

$ sudo chown newuser:newgroup file.txt
$ ls -l file.txt
-rw-r--r--  1 newuser newgroup  50 Sep 23 11:15 file.txt

$ chgrp developers project.txt
$ ls -l project.txt
-rw-r--r--  1 user developers  64 Sep 23 11:16 project.txt
```

Exercise 5: File Searching with `find`

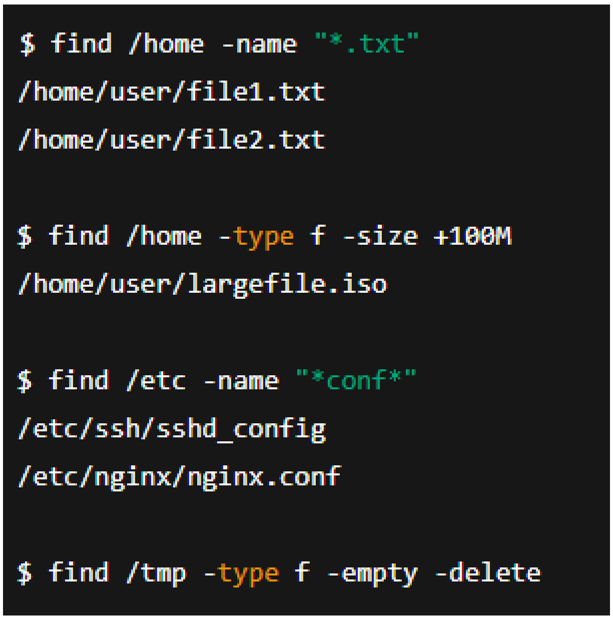
Task Statement:

Search files by name, type, size, and permissions using `find`.

Command(s):

```
find /home -name "*.txt"
find /home -type f -size +100M
find /etc -name "*conf*"
find /tmp -type f -empty -delete
```

Output:



```
$ find /home -name "*.txt"
/home/user/file1.txt
/home/user/file2.txt

$ find /home -type f -size +100M
/home/user/largefile.iso

$ find /etc -name "*conf*"
/etc/ssh/sshd_config
/etc/nginx/nginx.conf

$ find /tmp -type f -empty -delete
```

Exercise 6: Pattern Searching with `grep`

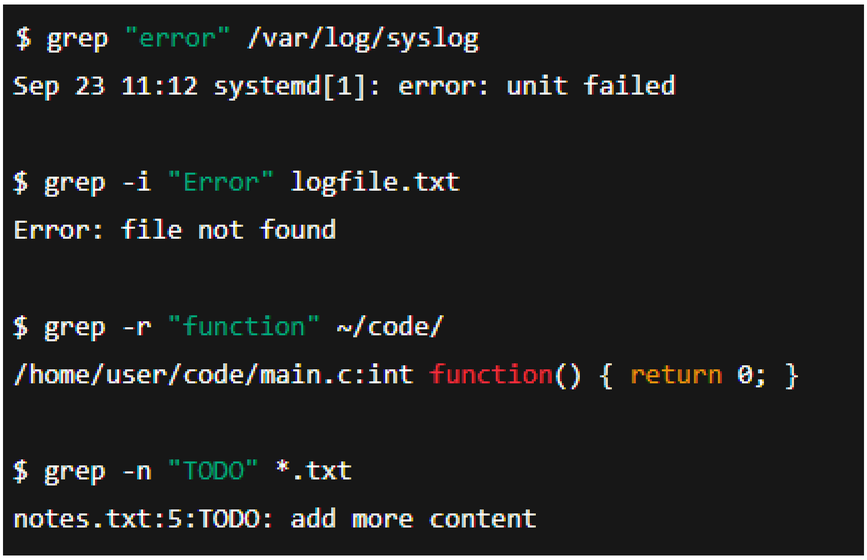
Task Statement:

Search for patterns in files using `grep`.

Command(s):

```
grep "error" /var/log/syslog
grep -i "Error" logfile.txt
grep -r "function" ~/code/
grep -n "TODO" *.txt
```

Output:



```
$ grep "error" /var/log/syslog
Sep 23 11:12 systemd[1]: error: unit failed

$ grep -i "Error" logfile.txt
Error: file not found

$ grep -r "function" ~/code/
/home/user/code/main.c:int function() { return 0; }

$ grep -n "TODO" *.txt
notes.txt:5:TODO: add more content
```

Exercise 7: Archiving and Compression

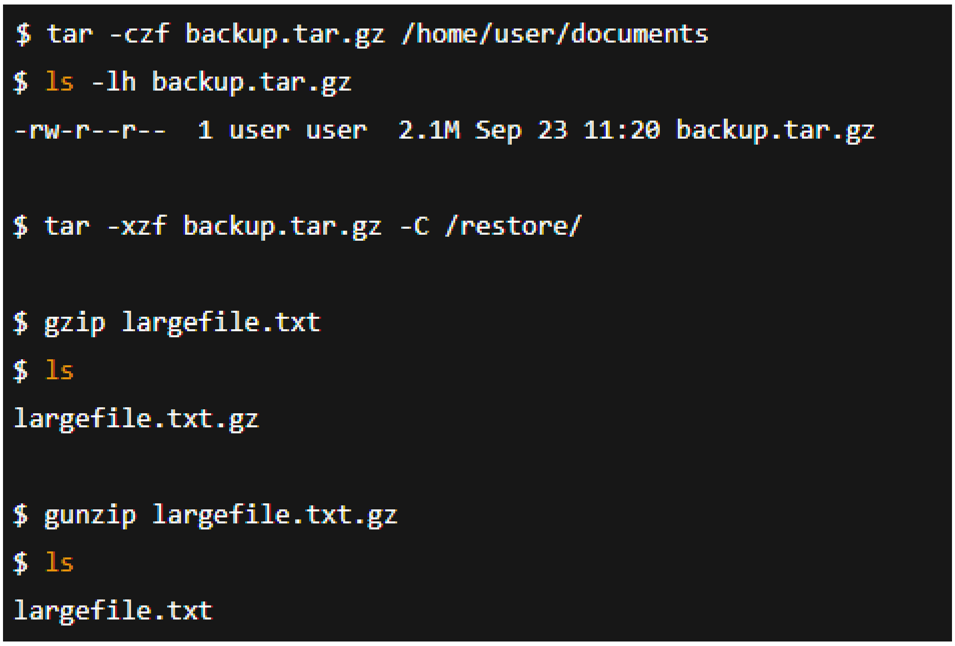
Task Statement:

Create and extract archives using `tar`, compress and decompress with `gzip`/`gunzip`.

Command(s):

```
tar -czf backup.tar.gz /home/user/documents
tar -xzf backup.tar.gz -C /restore/
gzip largefile.txt
gunzip largefile.txt.gz
```

Output:



```
$ tar -czf backup.tar.gz /home/user/documents
$ ls -lh backup.tar.gz
-rw-r--r--  1 user user  2.1M Sep 23 11:20 backup.tar.gz

$ tar -xzf backup.tar.gz -C /restore/

$ gzip largefile.txt
$ ls
largefile.txt.gz

$ gunzip largefile.txt.gz
$ ls
largefile.txt
```

Exercise 8: Creating Links

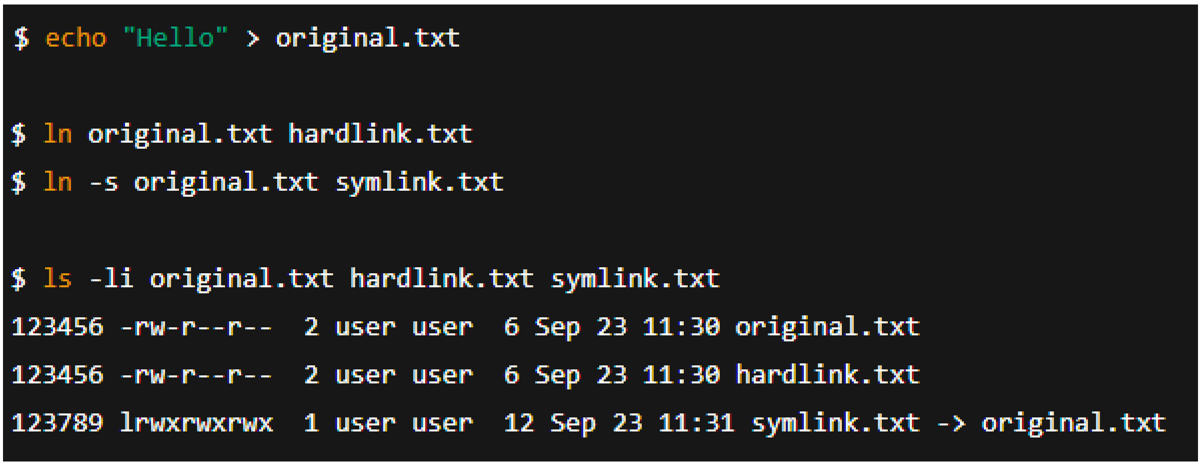
Task Statement:

Create and test hard and symbolic links using `ln`.

Command(s):

```
echo "Hello" > original.txt
ln original.txt hardlink.txt
ln -s original.txt symlink.txt
ls -li original.txt hardlink.txt symlink.txt
```

Output:



```
$ echo "Hello" > original.txt

$ ln original.txt hardlink.txt
$ ln -s original.txt symlink.txt

$ ls -li original.txt hardlink.txt symlink.txt
123456 -rw-r--r--  2 user user  6 Sep 23 11:30 original.txt
123456 -rw-r--r--  2 user user  6 Sep 23 11:30 hardlink.txt
123789 lrwxrwxrwx  1 user user  12 Sep 23 11:31 symlink.txt -> original.txt
```

Result

- Successfully created, copied, moved, and deleted files.
- Practiced viewing file contents and monitoring logs.
- Explored file permissions and ownership management.
- Used `find` and `grep` to locate and filter data.
- Created archives and compressed files.
- Demonstrated both hard and symbolic links.

Challenges Faced & Learning Outcomes

- Challenge 1: Accidentally deleted files with `rm` without `-i`. Learned to use `rm -i` for safety.
- Challenge 2: Remembering numeric vs symbolic permissions in `chmod`. Fixed through repeated practice.

Learning:

- Gained practical skills with file manipulation and permission commands.
- Learned how to efficiently search files and patterns in Linux.
- Understood how to archive and compress files for better storage management.
- Understood differences between hard and symbolic links.

Conclusion

This experiment provided hands-on experience with core Linux file management, permissions, searching, archiving, and linking. These are foundational skills for effective Linux system administration and daily usage.