

# Day 10 – File Permissions & File Operations Challenge

## 🎯 Objective

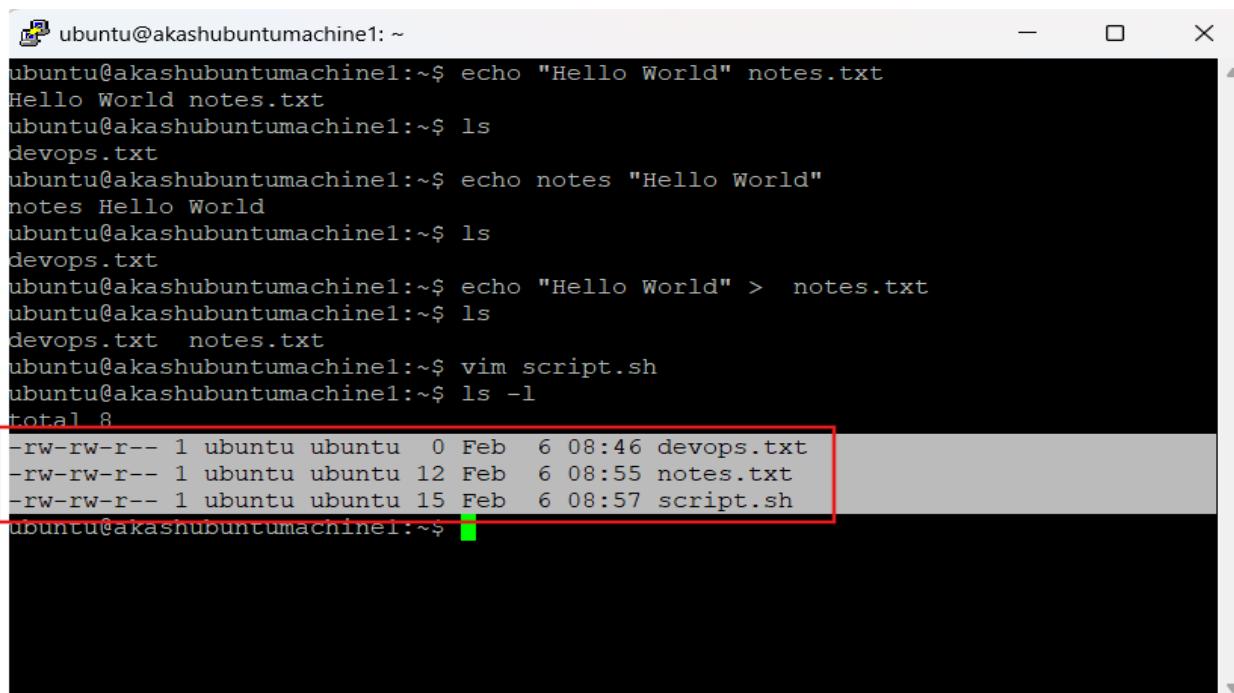
Master file permissions and basic file operations in Linux.

You will practice:

- Creating files using touch, cat, vim
  - Reading files using cat, head, tail
  - Understanding Linux permission format
  - Modifying permissions using chmod
  - Testing permission behavior
- 

## 📝 Task 1: Create Files

1. Create empty file devops.txt using touch
2. Create notes.txt with some content using cat or echo
3. Create script.sh using vim with content: echo "Hello DevOps"



The screenshot shows a terminal window with a black background and white text. It displays a series of commands entered by the user and their corresponding outputs. The user has created three files: 'devops.txt', 'notes.txt', and 'script.sh'. The 'ls -l' command at the bottom shows the details of these files, including their permissions (rw-rw-r--), owner (ubuntu), group (ubuntu), modification date (Feb 12, 15), and creation time (08:46, 08:55, 08:57 respectively). A red box highlights the output of the 'ls -l' command.

```
ubuntu@akashubuntumachine1: ~
ubuntu@akashubuntumachine1:~$ echo "Hello World" > notes.txt
Hello World
ubuntu@akashubuntumachine1:~$ ls
devops.txt
ubuntu@akashubuntumachine1:~$ echo notes "Hello World"
notes Hello World
ubuntu@akashubuntumachine1:~$ ls
devops.txt
ubuntu@akashubuntumachine1:~$ echo "Hello World" > notes.txt
ubuntu@akashubuntumachine1:~$ ls
devops.txt notes.txt
ubuntu@akashubuntumachine1:~$ vim script.sh
ubuntu@akashubuntumachine1:~$ ls -l
total 8
-rw-rw-r-- 1 ubuntu ubuntu 0 Feb  6 08:46 devops.txt
-rw-rw-r-- 1 ubuntu ubuntu 12 Feb  6 08:55 notes.txt
-rw-rw-r-- 1 ubuntu ubuntu 15 Feb  6 08:57 script.sh
ubuntu@akashubuntumachine1:~$
```

## Task 2: Read Files

1. Read notes.txt using cat
2. View script.sh in vim read-only mode
3. Display first 5 lines of /etc/passwd using head
4. Display last 5 lines of /etc/passwd using tail

```
ubuntu@akashubuntumachine1: ~
ubuntu@akashubuntumachine1:~$ cat notes.txt
Hello World
ubuntu@akashubuntumachine1:~$ ls -l
total 8
-rw-rw-r-- 1 ubuntu ubuntu 0 Feb  6 08:46 devops.txt
-rw-rw-r-- 1 ubuntu ubuntu 12 Feb  6 08:55 notes.txt
-rw-rw-r-- 1 ubuntu ubuntu 15 Feb  6 08:57 script.sh
ubuntu@akashubuntumachine1:~$ chown 444 script.sh
chown: changing ownership of 'script.sh': Operation not permitted
ubuntu@akashubuntumachine1:~$ ls -l
total 8
-rw-rw-r-- 1 ubuntu ubuntu 0 Feb  6 08:46 devops.txt
-rw-rw-r-- 1 ubuntu ubuntu 12 Feb  6 08:55 notes.txt
-rw-rw-r-- 1 ubuntu ubuntu 15 Feb  6 08:57 script.sh
ubuntu@akashubuntumachine1:~$ chmod 444 script.sh
ubuntu@akashubuntumachine1:~$ ls -l
total 8
-rw-rw-r-- 1 ubuntu ubuntu 0 Feb  6 08:46 devops.txt
-rw-rw-r-- 1 ubuntu ubuntu 12 Feb  6 08:55 notes.txt
-r--r--r-- 1 ubuntu ubuntu 15 Feb  6 08:57 script.sh
ubuntu@akashubuntumachine1:~$ vim script.sh
ubuntu@akashubuntumachine1:~$ tail -5 /etc/passwd
pcp:x:997:988:Performance Co-Pilot:/var/lib/pcp:/usr/sbin/nologin
tokyo:x:1003:1003::/home/tokyo:/bin/sh
berlin:x:1004:1004::/home/berlin:/bin/sh
professor:x:1005:1005::/home/professor:/bin/sh
nairobi:x:1001:1001,,,,:/home/nairobi:/bin/bash
ubuntu@akashubuntumachine1:~$ head -5 /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
ubuntu@akashubuntumachine1:~$ █
```

### Task 3: Understand Permissions

**Format: rwxrwxrwx (owner-group-others)**

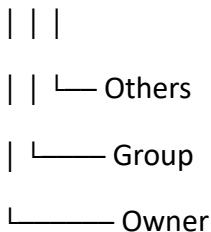
- r = read (4), w = write (2), x = execute (1)

**Check your files: ls -l devops.txt notes.txt script.sh**

**Answer: What are current permissions? Who can read/write/execute?**

#### **Permission Format**

rwxrwxrwx



#### **Values:**

- r = read (4)
- w = write (2)
- x = execute (1)

### Check Permissions

**ls -l devops.txt notes.txt script.sh**

### Observations:

File	Permissions	Owner	Group	Others
devops.txt	rw-r--r--	Read/Write	Read	Read
notes.txt	rw-r--r--	Read/Write	Read	Read
script.sh	rw-r--r--	Read/Write	Read	Read

## Task 4: Modify Permissions

1. Make `script.sh` executable → run it with `./script.sh`
2. Set `devops.txt` to read-only (remove write for all)
3. Set `notes.txt` to 640 (owner: rw, group: r, others: none)
4. Create directory `project/` with permissions 755

```
ubuntu@akashubuntumachine1: ~
ubuntu@akashubuntumachine1:~$ ls -l
total 8
-rw-rw-r-- 1 ubuntu ubuntu 0 Feb 6 08:46 devops.txt
-rw-rw-r-- 1 ubuntu ubuntu 12 Feb 6 08:55 notes.txt
-r--r--r-- 1 ubuntu ubuntu 15 Feb 6 08:57 script.sh
ubuntu@akashubuntumachine1:~$ chmod +x script.sh
ubuntu@akashubuntumachine1:~$ ls -l
total 8
-rw-rw-r-- 1 ubuntu ubuntu 0 Feb 6 08:46 devops.txt
-rw-rw-r-- 1 ubuntu ubuntu 12 Feb 6 08:55 notes.txt
-r-xr-xr-x 1 ubuntu ubuntu 15 Feb 6 08:57 script.sh
ubuntu@akashubuntumachine1:~$ chmod +r devops.txt
ubuntu@akashubuntumachine1:~$ ls -l
total 8
-rw-rw-r-- 1 ubuntu ubuntu 0 Feb 6 08:46 devops.txt
-rw-rw-r-- 1 ubuntu ubuntu 12 Feb 6 08:55 notes.txt
-r-xr-xr-x 1 ubuntu ubuntu 15 Feb 6 08:57 script.sh
ubuntu@akashubuntumachine1:~$ chmod 444 devops.txt
ubuntu@akashubuntumachine1:~$ ls -l
total 8
-r--r--r-- 1 ubuntu ubuntu 0 Feb 6 08:46 devops.txt
-rw-r--r-- 1 ubuntu ubuntu 12 Feb 6 08:55 notes.txt
-r-xr-xr-x 1 ubuntu ubuntu 15 Feb 6 08:57 script.sh
ubuntu@akashubuntumachine1:~$ chmod 640 notes.txt
ubuntu@akashubuntumachine1:~$ ls -l
total 8
-r--r--r-- 1 ubuntu ubuntu 0 Feb 6 08:46 devops.txt
-rw-r----- 1 ubuntu ubuntu 12 Feb 6 08:55 notes.txt
-r-xr-xr-x 1 ubuntu ubuntu 15 Feb 6 08:57 script.sh
ubuntu@akashubuntumachine1:~$ mkdir project
ubuntu@akashubuntumachine1:~$ ls -l
total 12
-r--r--r-- 1 ubuntu ubuntu 0 Feb 6 08:46 devops.txt
-rw-r----- 1 ubuntu ubuntu 12 Feb 6 08:55 notes.txt
drwxrwxr-x 2 ubuntu ubuntu 4096 Feb 6 09:20 project
-r-xr-xr-x 1 ubuntu ubuntu 15 Feb 6 08:57 script.sh
ubuntu@akashubuntumachine1:~$ chmod 755 project/
ubuntu@akashubuntumachine1:~$ ls -l
total 12
-r--r--r-- 1 ubuntu ubuntu 0 Feb 6 08:46 devops.txt
-rw-r----- 1 ubuntu ubuntu 12 Feb 6 08:55 notes.txt
drwxr-xr-x 2 ubuntu ubuntu 4096 Feb 6 09:20 project
-r-xr-xr-x 1 ubuntu ubuntu 15 Feb 6 08:57 script.sh
ubuntu@akashubuntumachine1:~$
```

 **Make script executable**

```
chmod +x script.sh
```

```
./script.sh
```

 **Set devops.txt to read-only**

```
chmod 444 devops.txt
```

 **Set notes.txt to 640**

```
chmod 640 notes.txt
```

 **Create directory with 755 permissions**

```
mkdir project
```

```
chmod 755 project
```

 **Verify**

```
ls -l
```

 **Screenshot:**

(Attach screenshot showing permission changes)

---

 **Task 5: Test Permissions**

 **Try writing to read-only file**

```
echo "test" >> devops.txt
```

**Result:**

Permission denied 

---

 **Try executing file without execute permission**

```
chmod -x script.sh
```

```
./script.sh
```

**Result:**

Permission denied 

---

## Learning Outcomes

- Understood Linux permission structure (rwx)
  - Practiced chmod with numeric & symbolic modes
  - Learned how permissions affect file execution and editing
  - Gained hands-on experience with real Linux file operations
- 

## Conclusion

File permissions are a critical part of Linux security.

This challenge helped build practical understanding of:

- Who can read/write/execute files
- How to control access using chmod
- How Linux enforces security at file level