

Part 4

Searching and Filtering Linux

In Linux you search two different things :

1. Text inside files → `grep`
2. Files themselves → `find`

If you remember just that, you're already oriented.

Section 1 : `grep` — Search Text inside Files

What `grep` Does ?

`grep` looks line by line and prints only the lines that match a pattern.

```
grep "error" logfile.txt
```

Meaning :

- Open `logfile.txt`
- Show lines containing error

Case-insensitive search:

```
grep -i "error" logfile.txt
```

Matches :

- error
- ERROR
- Error

Very common in logs.

Search across many files :

```
grep "main" *.c
```

Search “main” in all `.c` files.

Recursive Search (Folder inside Folder) :

```
grep -r "password" /etc
```

Requires permission.

Show line numbers (useful for Debugging) :

```
grep -n "fail" file.txt
```

Invert search (exclude matches) :

```
grep -v "info" logfile.txt
```

shows everything except “info”.

SECTION 2: `grep` + Pipes (Very Powerful)

```
ps aux | grep nginx
```

Meaning:

- List processes
- Filter nginx only

```
dmesg | grep -i usb
```

Check USB-related system messages.

SECTION 3: `find` — Search Files & Folders

What `find` Does

`find` searches **file names and properties**, not content.

```
find . -name "data.txt"
```

Meaning:

- Start from current directory
- Look for `data.txt`

Find by extension

```
find /var/log -name "*.log"
```

Find large files

```
find /home -size +100M
```

Files bigger than 100MB.

Find by time (very practical)

```
find . -mtime -7
```

Modified in last 7 days.

Option	Meaning
<code>-mtime -7</code>	Last 7 days
<code>-mtime +7</code>	Older than 7 days

SECTION 4: Combine `find` with Actions

Delete files safely

```
find . -name "*.tmp" -delete
```

Always test first:

```
find . -name "*.tmp"
```

Execute command on each result

```
find . -name "*.sh" -exec chmod +x {} \;
```

Meaning:

- Find shell scripts
- Make them executable

`{}` = found file

`\;` = end command

SECTION 5: `grep` vs `find` (Important Comparison)

Task	Tool
Search text inside files	<code>grep</code>
Search files by name/size/date	<code>find</code>
Search logs	<code>grep</code>
Cleanup files	<code>find</code>

SECTION 6: Real-World Professional Examples

1. Find failed logins

```
grep "Failed" /var/log/auth.log
```

2. Find config files

```
find /etc -name "*.conf"
```

3. Search errors in last 24 hours

```
find /var/log -mtime -1 -exec grep "error" {} \;
```

4. Count matches

```
grep -r "TODO" . | wc -l
```

SECTION 7: Common Beginner Mistakes

Using `find` to search text

Using `grep` to search file names

Remember:

`grep` = inside files

`find` = files themselves