

Day 2 – Phase 2: File & Directory Management + Search

Boss's Request: Organize project files and simulate sensor config checks.

Tasks:

- Inside `iot_logger`, create `logs/temperature.log` and `scripts/sensor_script.py`.

```
cd iot_logger
touch logs/temperature.log
touch scripts/sensor_script.py
```

```
mohamed@iot ~> cd iot_logger/
mohamed@iot ~/iot_logger> touch logs/temperature.log
mohamed@iot ~/iot_logger> touch scripts/sensor_script.py
mohamed@iot ~/iot_logger> tree .
.
├── data
├── logs
│   └── temperature.log
└── scripts
    └── sensor_script.py
3 directories, 2 files
```

- Copy `/etc/services` into `data` and search for patterns like `ssh` or `http`.

```
cp /etc/services data
grep ssh data/services
grep http data/services
```

```
mohamed@iot ~/iot_logger> cp /etc/services data
mohamed@iot ~/iot_logger> grep ssh data/services
ssh                22/tcp             # SSH Remote Login Protocol
mohamed@iot ~/iot_logger> grep http data/services
# Updated from https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml
http                80/tcp             # WorldWideWeb HTTP
https               443/tcp            # http protocol over TLS/SSL
https               443/udp            # HTTP/3
http-alt            8080/tcp            webcache           # WWW caching service
```

- Use regex to find lines starting with `t` or containing **numbers**.

```
grep ^t data/services
grep "[0-9]" data/services # we can use less (grep "[0-9]" data/services |
less)

# or
grep -e "^t" -e "[0-9]" data/services
```

```
mohamed@iot ~/iot_logger> grep "^t" data/services
tcpmux      1/tcp          # TCP port service multiplexer
telnet      23/tcp
time        37/tcp          timserver
time        37/udp          timserver
tacacs      49/tcp          # Login Host Protocol (TACACS)
tacacs      49/udp
tftp        69/udp
talk        517/udp
tinc        655/tcp          # tinc control port
tinc        655/udp
telnet      992/tcp          # Telnet over SSL
tproxy      8081/tcp         # Transparent Proxy
tfido       60177/tcp        # fidonet EMSI over telnet
mohamed@iot ~/iot_logger> grep "[0-9]" data/services
tcpmux      1/tcp          # TCP port service multiplexer
echo        7/tcp
echo        7/udp
discard     9/tcp          sink null
discard     9/udp          sink null
sysstat     11/tcp         users
daytime     13/tcp
daytime     13/udp
netstat     15/tcp
qotd        17/tcp          quote
```

- Locate `.txt` files in `/home/<username>` and remove temporary ones if needed.

```
find /home/$USER -name "*.txt"
# if wanted to remove use rm
rm /home/mohamed/iot/gitdemo/file.txt
```

```
mohamed@iot ~/iot_logger> find /home/$USER -name "*.txt"
/home/mohamed/.cache/tracker3/files/no-need-mtime-check.txt
/home/mohamed/.cache/tracker3/files/locale-for-miner-apps.txt
/home/mohamed/.cache/tracker3/files/last-crawl.txt
/home/mohamed/.cache/tracker3/files/first-index.txt
/home/mohamed/snap/firefox/common/.mozilla/firefox/ttau78t2.default/pkcs11.txt
/home/mohamed/iot/gitdemo/file.txt
/home/mohamed/iot/gitdemo/test.txt
mohamed@iot ~/iot_logger> rm /home/mohamed/iot/gitdemo/file.txt
```

- Create **hard** and **symbolic** links for `temperature.log` .

```
# hard link
ln ./logs/temperature.log hard_temperature.log
# symbolic link
ln -s ./logs/temperature.log symbolic_temperature.log
```

```
mohamed@iot ~/iot_logger> cd logs/
mohamed@iot ~/i/logs> ln temperature.log hard_temperature.log
mohamed@iot ~/i/logs> ln -s temperature.log symbolic_temperature.log
mohamed@iot ~/i/logs> ls -l
total 0
-rw-rw-r-- 2 mohamed mohamed 0 Aug 31 21:23 hard_temperature.log
lrwxrwxrwx 1 mohamed mohamed 15 Aug 31 21:53 symbolic_temperature.log -> temperature.log
-rw-rw-r-- 2 mohamed mohamed 0 Aug 31 21:23 temperature.log
```

- Display directory structure to confirm organization.

```
tree ~/iot_logger
```

```
mohamed@iot ~/i/logs> tree ~/iot_logger/
/home/mohamed/iot_logger/
├── data
│   └── services
├── logs
│   ├── hard_temperature.log
│   ├── symbolic_temperature.log -> temperature.log
│   └── temperature.log
└── scripts
    └── sensor_script.py

3 directories, 5 files
```

Open-Ended Questions:

- Explain the different types of files in Linux (regular, directory, symbolic link, device, etc.) and how to check them with commands.

Type	Symbol	Explain
Regular	-	file (txt, binary,...)
Directory	d	Folder
Symbolic	l	pointer to another file (shortcut)
Device	b or c	block (disk) or character (keyboard)

- check them with:
 - `ls -l`
 - `file <file_name>`
- What's the difference between a **hard link** and a **symbolic link**? Give real examples of when to use each.
 - **Hard Link** →
 - Another name for the same **inode**
 - File exists even if original is deleted
 - **Symbolic Link** →
 - **Pointer (shortcut)** to original path
 - Breaks if original is delete.
 - Examples:
 - **Hard link** for backup-like copies
 - `/bin` → `/usr/bin` (To keep old programs working without duplicating files) (& easy)
- Is `rmdir` the same as `rm -r` when deleting directories? Explain.
 - `rmdir` → removes empty directories only
 - `rm -r` → removes directory and its contents recursively