SIC-IOT702 / task 03 : capstone Task / phase 2

Instractor : eng/ Mohamed Ahmed

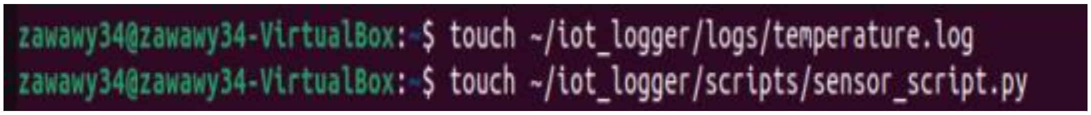
Student name: Ahmed Mohamed Elsayed

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

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

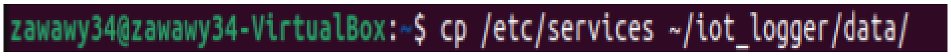
Tasks:

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

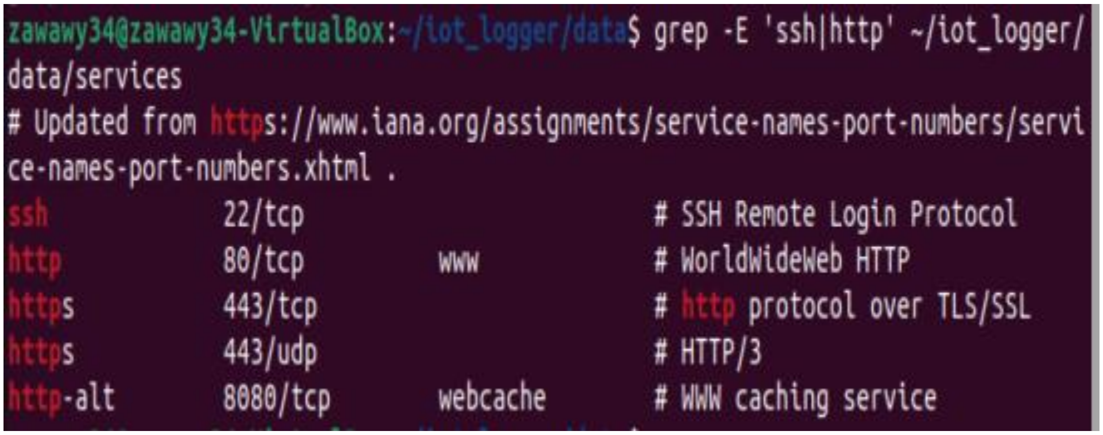


```
zawawy34@zawawy34-VirtualBox:~$ touch ~/iot_logger/logs/temperature.log
zawawy34@zawawy34-VirtualBox:~$ touch ~/iot_logger/scripts/sensor_script.py
```

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

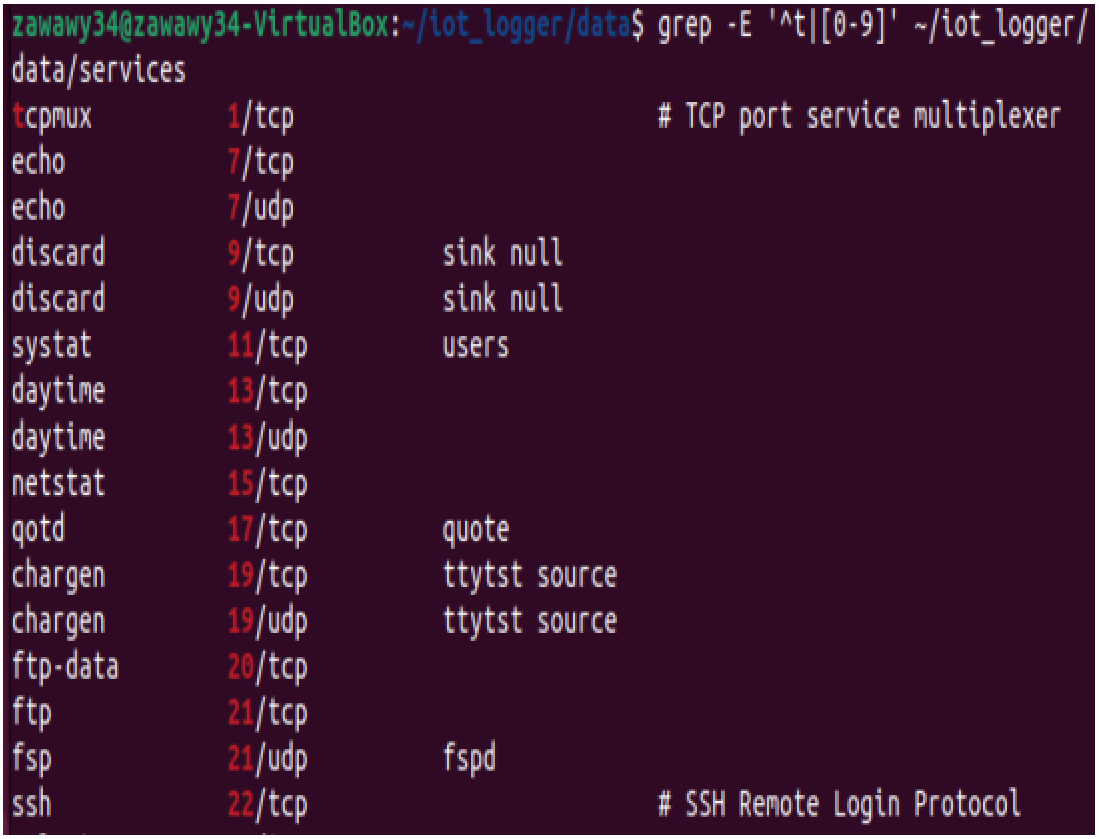


```
zawawy34@zawawy34-VirtualBox:~$ cp /etc/services ~/iot_logger/data/
```



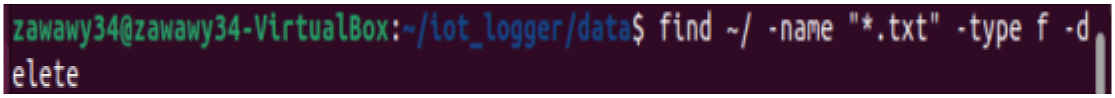
```
zawawy34@zawawy34-VirtualBox:~/iot_logger/data$ grep -E 'ssh|http' ~/iot_logger/
data/services
# Updated from https://www.iana.org/assignments/service-names-port-numbers/servi
ce-names-port-numbers.xhtml .
ssh             22/tcp                          # SSH Remote Login Protocol
http            80/tcp          www             # 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.



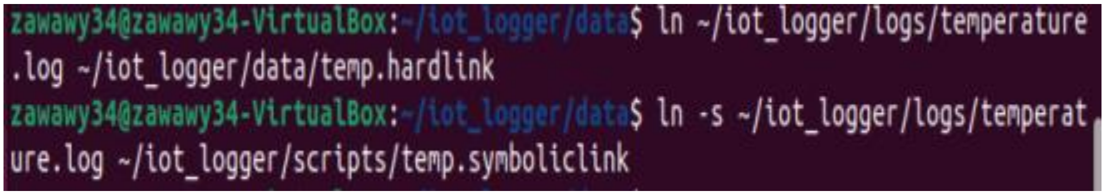
```
zawawy34@zawawy34-VirtualBox:~/iot_logger/data$ grep -E '^t|[0-9]' ~/iot_logger/
data/services
tcpmux          1/tcp                           # TCP port service multiplexer
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
daytime         13/tcp
daytime         13/udp
netstat         15/tcp
qotd            17/tcp          quote
chargen         19/tcp          ttytst source
chargen         19/udp          ttytst source
ftp-data        20/tcp
ftp             21/tcp
fsp             21/udp          fspd
ssh             22/tcp                          # SSH Remote Login Protocol
```

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



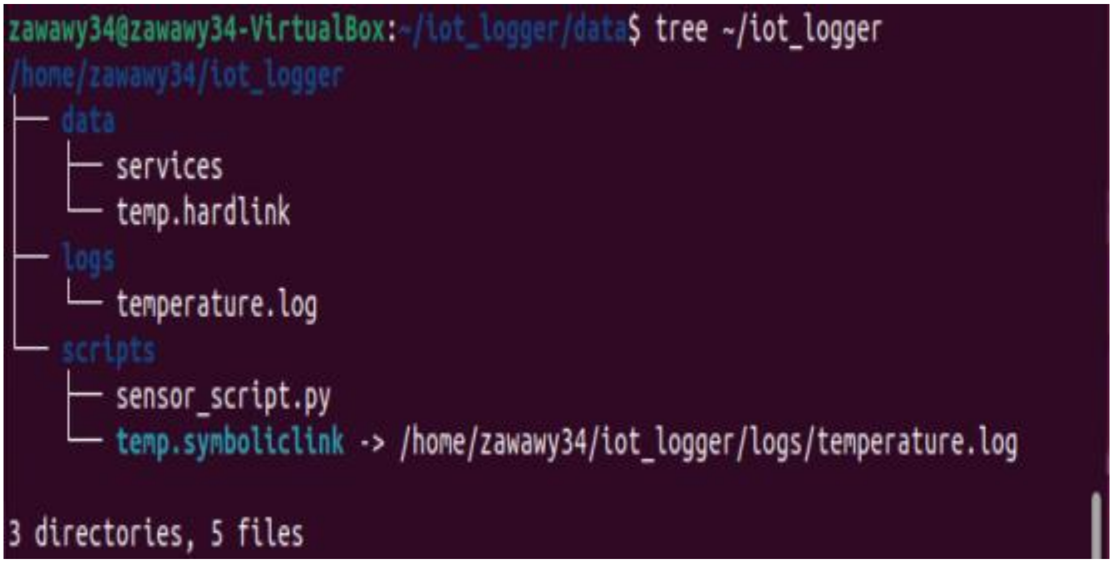
```
zawawy34@zawawy34-VirtualBox:~/iot_logger/data$ find ~/ -name "*.txt" -type f -d
elete
```

- Create hard and symbolic links for temperature.log.



```
zawawy34@zawawy34-VirtualBox:~/iot_logger/data$ ln ~/iot_logger/logs/temperature
.log ~/iot_logger/data/temp.hardlink
zawawy34@zawawy34-VirtualBox:~/iot_logger/data$ ln -s ~/iot_logger/logs/temperat
ure.log ~/iot_logger/scripts/temp.symboliclink
```

- Display directory structure to confirm organization.



```
zawawy34@zawawy34-VirtualBox:~/iot_logger/data$ tree ~/iot_logger
/home/zawawy34/iot_logger
├── data
│   ├── services
│   └── temp.hardlink
├── logs
│   └── temperature.log
└── scripts
    ├── sensor_script.py
    └── temp.symboliclink -> /home/zawawy34/iot_logger/logs/temperature.log

3 directories, 5 files
```

Open-Ended Questions:

1 Types of Files in Linux:

- Linux treats everything as a file, but there are different types. You can check a file's type using the

ls -l command, which shows the file type in the first character of the permissions string.

- **Regular File (-)**: A standard file containing data, such as a text document, image, or executable program.
- **Directory (d)**: A file that contains a list of other files and directories.
- **Symbolic Link (l)**: A pointer or shortcut to another file or directory.
- **Device File (b or c)**: Represents a physical device, like a hard drive (b for block device) or a serial port (c for character device).
- **Socket (s)**: A special file used for inter-process communication (IPC) over a network.
- **Named Pipe (p)**: A special file used for IPC within the same system.

2 Hard Link vs. Symbolic Link:

- A **hard link** is a direct reference to the file's data on the disk (its inode). It points to the same physical data as the original file. All hard links to a file are equally valid, and the file's data is only deleted when all hard links are removed. You cannot create a hard link for a directory or across different file systems.

  - **Example Use Case**: A project might need to access a shared configuration file from multiple locations. Creating a hard link ensures that any changes to the file are immediately reflected in all locations because they all refer to the same data.

- A **symbolic link** (or soft link) is a separate file that contains the path to the original file. It is a shortcut. If the original file is deleted, the symbolic link will become "broken" or "dangling," as it points to a nonexistent path. Symbolic links can link to directories and span different file systems.

  - **Example Use Case**: A web server needs to access logs stored in a different, deeply nested directory. A symbolic link can be created in the web root directory to easily access the log directory without moving the original files. This is common for creating simple, user-friendly paths to complex file locations.

3 rmdir vs. rm -r:

- rmdir is used to remove an **empty** directory. It will fail if the directory contains any files or subdirectories.

- rm -r (or rm -R) is a more powerful and dangerous command used to remove a directory and its **entire contents recursively**. It deletes all files and subdirectories within the specified directory. This command should be used with caution as it permanently removes data without a trash or recycle bin. The -r flag stands for "recursive."