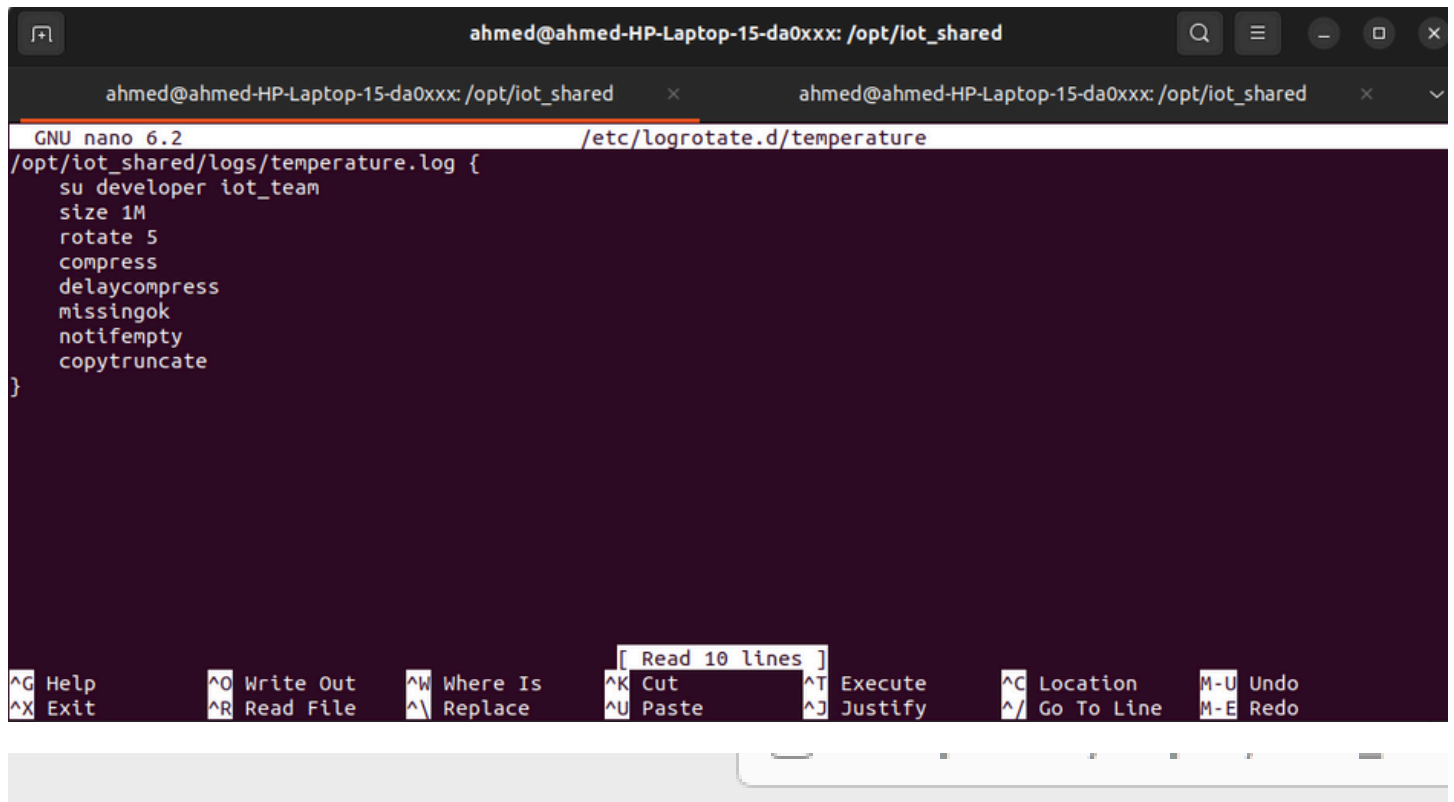# Name : Ahmed Khaled Abdelmaksod

Group : IOT_701_O

DAY6

- Configure log rotation for temperature.log (rotate at 1 MB, compress).
- Test by forcing a rotation.

```
  GNU nano 6.2                          /etc/logrotate.d/temperature
/opt/iot_shared/logs/temperature.log {
    su developer iot_team
    size 1M
    rotate 5
    compress
    delaycompress
    missingok
    notifempty
    copytruncate
}




                                    [ Read 10 lines ]
^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo
```

temperature.log   temperature.log.1

```
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$ sudo logrotate -v /etc/logrotate.d/temperature
reading config file /etc/logrotate.d/temperature
Reading state from file: /var/lib/logrotate/status
Allocating hash table for state file, size 64 entries
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
Creating new state
```

```
Handling 1 logs

rotating pattern: /opt/iot_shared/logs/temperature.log  1048576 bytes (5 rotations)
empty log files are not rotated, old logs are removed
switching euid from 0 to 1001 and egid from 0 to 1001 (pid 10942)
considering log /opt/iot_shared/logs/temperature.log
  Now: 2025-09-05 17:13
  Last rotated at 2025-09-05 16:00
  log needs rotating
rotating log /opt/iot_shared/logs/temperature.log, log->rotateCount is 5
dateext suffix '-20250905'
glob pattern '-[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]'
previous log /opt/iot_shared/logs/temperature.log.1 does not exist
renaming /opt/iot_shared/logs/temperature.log.5.gz to /opt/iot_shared/logs/temperature.log.6.gz (rotatecount 5, logs
tart 1, i 5),
old log /opt/iot_shared/logs/temperature.log.5.gz does not exist
renaming /opt/iot_shared/logs/temperature.log.4.gz to /opt/iot_shared/logs/temperature.log.5.gz (rotatecount 5, logs
tart 1, i 4),
old log /opt/iot_shared/logs/temperature.log.4.gz does not exist
renaming /opt/iot_shared/logs/temperature.log.3.gz to /opt/iot_shared/logs/temperature.log.4.gz (rotatecount 5, logs
tart 1, i 3),
old log /opt/iot_shared/logs/temperature.log.3.gz does not exist
renaming /opt/iot_shared/logs/temperature.log.2.gz to /opt/iot_shared/logs/temperature.log.3.gz (rotatecount 5, logs
```

```
previous log /opt/iot_shared/logs/temperature.log.1 does not exist
renaming /opt/iot_shared/logs/temperature.log.5.gz to /opt/iot_shared/logs/temperature.log.6.gz (rotatecount 5, logs
tart 1, i 5),
old log /opt/iot_shared/logs/temperature.log.5.gz does not exist
renaming /opt/iot_shared/logs/temperature.log.4.gz to /opt/iot_shared/logs/temperature.log.5.gz (rotatecount 5, logs
tart 1, i 4),
old log /opt/iot_shared/logs/temperature.log.4.gz does not exist
renaming /opt/iot_shared/logs/temperature.log.3.gz to /opt/iot_shared/logs/temperature.log.4.gz (rotatecount 5, logs
tart 1, i 3),
old log /opt/iot_shared/logs/temperature.log.3.gz does not exist
renaming /opt/iot_shared/logs/temperature.log.2.gz to /opt/iot_shared/logs/temperature.log.3.gz (rotatecount 5, logs
tart 1, i 2),
old log /opt/iot_shared/logs/temperature.log.2.gz does not exist
renaming /opt/iot_shared/logs/temperature.log.1.gz to /opt/iot_shared/logs/temperature.log.2.gz (rotatecount 5, logs
tart 1, i 1),
old log /opt/iot_shared/logs/temperature.log.1.gz does not exist
renaming /opt/iot_shared/logs/temperature.log.0.gz to /opt/iot_shared/logs/temperature.log.1.gz (rotatecount 5, logs
tart 1, i 0),
old log /opt/iot_shared/logs/temperature.log.0.gz does not exist
log /opt/iot_shared/logs/temperature.log.6.gz doesn't exist -- won't try to dispose of it
copying /opt/iot_shared/logs/temperature.log to /opt/iot_shared/logs/temperature.log.1
truncating /opt/iot_shared/logs/temperature.log
switching euid from 1001 to 0 and egid from 1001 to 0 (pid 10942)
```

- **Schedule the Python script to run every 5 minutes with cron.**
- **Verify log growth over time.**

```
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$ crontab -e
no crontab for ahmed - using an empty one

Select an editor.  To change later, run 'select-editor'.
  1. /bin/nano        <---- easiest
  2. /usr/bin/vim.basic
  3. /usr/bin/vim.tiny
  4. /usr/bin/code
  5. /bin/ed

Choose 1-5 [1]: 1
crontab: installing new crontab
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$ crontab -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
```

before 5 min the size of log is 356kb

```
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
*/5 * * * * /usr/bin/python3 /opt/iot_shared/scripts/adjusted_sensor.py >> /opt/iot_shared/logs/temperature.log 2>&1
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$ ls -lh /opt/iot_shared/logs/
total 1.8M
-rw-rw-r-- 1 developer iot_team 356K Sep  5 17:32 temperature.log
-rw-rw-r-- 1 developer iot_team 1.4M Sep  5 17:13 temperature.log.1
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$
```

after 5 min the size of log is 383kb

```
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$ ls -lh /opt/iot_shared/logs/
total 1.8M
-rw-rw-r-- 1 developer iot_team 356K Sep  5 17:32 temperature.log
-rw-rw-r-- 1 developer iot_team 1.4M Sep  5 17:13 temperature.log.1
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$ ls -lh /opt/iot_shared/logs/
total 1.8M
-rw-rw-r-- 1 developer iot_team 383K Sep  5 17:35 temperature.log
-rw-rw-r-- 1 developer iot_team 1.4M Sep  5 17:13 temperature.log.1
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$
```

- **Compress old logs into .tar.gz in data/.**

```
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$ tar -czvf data/rotated_logs.tar.gz logs/temperature.log.*
logs/temperature.log.1
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$ ls -l data/
total 168
-rw-rw-r-- 1 ahmed ahmed 142533 Sep  5 17:44 rotated_logs.tar.gz
-rw-r--r-- 1 ahmed ahmed  12813 Aug 31 16:26 services
-rw-rw-r-- 1 ahmed ahmed   9604 Sep  5 15:38 temperature.log
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$
```

- **Simulate sending archives to /home/<username>/server/ using cp, scp, or rsync. (hint: use can use scp and copy to destination directory in another path on the same machine**

just for simulation).

```
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$ cp data/rotated_logs.tar.gz ~/server
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$ ls -l ~/server
-rw-rw-r-- 1 ahmed ahmed 142533 Sep  5 17:52 /home/ahmed/server
ahmed@ahmed-HP-Laptop-15-da0xxx:/opt/iot_shared$
```

Open-Ended Questions:

How does cron scheduling work? Show a crontab entry to run a script every 5 minutes.

Cron runs tasks automatically based on a time schedule defined in crontab.
Example: */5 * * * * /opt/iot_shared/scripts/sensor_script.py >> /opt/iot_shared/logs/temperature.log 2>&1

Why do we need log rotation? Show an example logrotate config for temperature.log.

Log rotation prevents logs from growing too large and consuming disk space.

/opt/iot_shared/logs/temperature.log {

    size 1M

    rotate 5

    compress

    missingok

    notifempty

}

Explain the difference between a Virtual Machine and a Container. Must containers use the same OS as the host? Why or why not?

VMs emulate full hardware and run separate OS kernels, while containers share the host kernel.
 Containers must use the same OS kernel as the host but can have different user-space libraries.

Reflection: Which actions in this project combined multiple Linux concepts (e.g.,redirection + process monitoring)? How does this apply to real IoT systems?

- We combined redirection, permissions, background jobs, process monitoring, and cron together.
- In IoT systems, these skills help automate data collection, ensure security, and manage resources efficiently.