Exp 4 (Periodic Processes Management & Logging)

Important to know:

- **الدقيقة (Minute)**: هي الدقيقة التي سيتم فيها تنفيذ الأمر. الخيارات المتاحة هي من 0 إلى 59.
- **الساعة (Hour)**: هي الساعة التي سيتم فيها تنفيذ الأمر. الخيارات المتاحة هي من 0 إلى 23.
- **اليوم من الشهر (DOM - Day of Month)**: هو اليوم من الشهر الذي سيتم فيه تنفيذ الأمر. الخيارات المتاحة هي من 0 إلى 31.
- **الشهر (Month)**: هو الشهر الذي سيتم فيه تنفيذ الأمر. الخيارات المتاحة هي من 1 إلى 12.
- **اليوم من الأسبوع (Week)**: هو اليوم من الأسبوع الذي سيتم فيه تنفيذ الأمر. الخيارات المتاحة هي من 0 إلى 6، حيث أن 0 يعني يوم الأحد.

3.1 Crontab Configuration

a. Make sure that Cron has been installed in your Linux machine. How can you do such check?

sudo service cron status

b. Install Cron if it does not exist. Show all the steps needed for installation.

sudo apt update

sudo apt install cron

c. Open the crontab file. What command do you use for this purpose?

crontab -e

d. Open the crontab using geditor. What do you need to do?

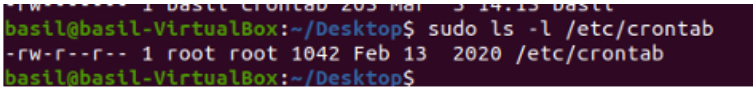
sudo gedit /etc/crontab

e. What does the command crontab -l do?

Is used to **list the current user's scheduled cron jobs**.

f. What permissions does the crontab file have?

sudo ls -l /etc/crontab



g. Which users have permissions to schedule cron tasks? Justify your answer?

بالنسبة لجدولة المهام باستخدام **cron**، إليك كيف الأمور بتكون:

1. **المستخدم الجذري (Root)**: المستخدم الجذري عنده صلاحيات كاملة، يعني يمكنه جدولة أي مهمة لأي مستخدم على النظام.
2. **المستخدمين في مجموعة خاصة**: في بعض الأنظمة، قد يكون هناك مجموعة خاصة (مثلاً مجموعة `cron` أو `crontab`) تُعطى صلاحيات جدولة المهام. المدير يمكنه تحديد هذه المجموعة عن طريق ملفات خاصة مثل `/etc/cron.allow` و `/etc/cron.deny` لتحديد من يحق له استخدام cron.
3. **المستخدمين العاديين**: عادةً، أي مستخدم عادي يقدر يحدد مهامه الخاصة باستخدام `crontab` إذا كانت الصلاحيات تسمح له بذلك. وموارد النظام مثل `/etc/cron.allow` أو `/etc/cron.deny` تتحكم في من يقدر يحدد مهامه.

السبب في ذلك:

- **الصلاحيات**: فقط المستخدمين اللي عندهم صلاحيات معينة في إعدادات النظام يقدروا يحددوا مهام cron. المستخدم الجذري عنده صلاحيات كاملة، بينما باقي المستخدمين ممكن يكونوا مقيدين أو مسموح لهم فقط بتحديد مهامهم الخاصة.
- **أمان النظام**: تحديد من يقدر يحدد مهام cron مهم للحفاظ على أمان النظام، لأن المهام يمكنها تنفيذ أوامر قد تكون حساسة أو لها صلاحيات عالية.

3.2 Implementing Periodic Tasks

a. Implement a periodic process that creates a file every 1 minute. The name of the file should be the time in which it will be created. Stop this process after 5 executions and show the files that have been created.

```
#!/bin/bash

COUNT_FILE="/home/wajdabdal-hadee/Desktop/count.txt"
dir="/home/wajdabdal-hadee/Desktop"

if [ ! -f "$COUNT_FILE" ] ; then
echo 0 > "$COUNT_FILE"
fi

COUNT=$(cat "$COUNT_FILE")

if [ "$COUNT" -lt 5 ] ; then
FILE_NAME="$(date '+%H-%M-%S').txt"
touch "$dir/$FILE_NAME"
```

```
echo "This file was created at $(date)" > "$dir/$FILE_NAME"
COUNT=$((COUNT + 1))
echo $COUNT > $COUNT_FILE
cat $COUNT_FILE
fi
```

Open | creatF.sh ~/Desktop | Save

```bash
#!/bin/bash

COUNT_FILE="/home/wajdabdal-hadee/Desktop/count.txt"
dir="/home/wajdabdal-hadee/Desktop"

if [ ! -f "$COUNT_FILE" ] ; then
    echo 0 > "$COUNT_FILE"
fi


COUNT=$(cat "$COUNT_FILE")


if [ "$COUNT" -lt 5 ] ; then
    FILE_NAME="$(date '+%H-%M-%S').txt"
    touch "$dir/$FILE_NAME"
    echo "This file was created at $(date)" > "$dir/$FILE_NAME"
    COUNT=$((COUNT + 1))
    echo $COUNT > $COUNT_FILE
    cat $COUNT_FILE
 fi
```

```
#
# m h  dom mon dow   command
* * * * * bash /home/wajdabdal-hadee/Desktop/creatF.sh
```

b. Modify the above task that you have created so that it runs every 2 minutes only on Sundays. Make sure that your process works fine and then stop it.

```
# m h  dom mon dow   command
*/2 * * * 0 bash /home/wajdabdal-hadee/Desktop/creatF.sh
```

c. Modify the above task that you have created so that it runs only on every system reboot. Make sure that your process works fine and then stop it.

on the crontab -e

@reboot date '+\%H:\%M:\%S' > /home/wajdabdal-hadee/Desktop/reboot.log

```
@reboot date '+\%H:\%M:\%S' >> /home/wajdabdal-hadee/Desktop/reboot.log
```

d. Write a periodic process that pings the gateway of your machine on every 12 am and 12 pm on everyday of the week. The ping should be logged into a file named ping.log under /var/log directory.

0 0,12 * * * ping -c 4 $(ip route | grep default | awk '{print $3}') >> /var/log/ping.log

```
0 0,12 * * * ping -c 4 $(ip route | grep default | awk '{print $3}') >> /home/wajdabdal-hadee/Desktop/ping.log
```

e. What does the command service cron status do?

service cron status

(e) ماذا يفعل الأمر `service cron status` ؟

- هذا الأمر يُظهر حالة خدمة cron، ويستخدم للتحقق مما إذا كانت تعمل:

```
service cron status
```

أو في بعض الأنظمة:

```
systemctl status cron
```

إذا لم تكن الخدمة تعمل، يمكن تشغيلها باستخدام:

```
sudo service cron start
```

3.3 Log files

a. Where can you find most of the log files in Linux?

most of the log files can be found in the /var/log directory.

This is the standard directory where system and application logs are stored.

b. Show some examples of log files in your machine. Is their naming convention consistent? Justify your answer?

```
basil@basil-VirtualBox:~/Desktop$ cd /var/log
basil@basil-VirtualBox:/var/log$ ls
alternatives.log    dmesg.4.gz          speech-dispatcher
alternatives.log.1  dpkg.log            syslog
apt                 dpkg.log.1          syslog.1
auth.log            faillog             ubuntu-advantage.log
boot.log            fontconfig.log      ubuntu-advantage-timer.log
boot.log.1          gdm3                ubuntu-advantage-timer.log.1
bootstrap.log       gpu-manager.log     unattended-upgrades
btmp                hp                  vboxadd-install.log
btmp.1              installer           vboxadd-setup.log
cups                journal             vboxadd-setup.log.1
dist-upgrade        kern.log            vboxadd-setup.log.2
dmesg               lastlog             vboxadd-setup.log.3
dmesg.0             oem-config.log      vboxadd-setup.log.4
dmesg.1.gz          openvpn             wtmp
dmesg.2.gz          ping.log            Xorg.0.log
dmesg.3.gz          private
```

c. What does the command lastlog do?

The **lastlog** command in Linux is used to display the most recent login information for all users on the system.

Example Output:

yaml Copy Edit

```yaml
Username         Port     From             Latest login
root             tty1     :0               Wed Mar  3 08:15:01 +03 2025
basil            pts/0    192.168.1.100    Wed Mar  3 08:17:24 +03 2025
```

d. What does log files rotation mean? Is it possible to implement it using cron? How?

Write it in crontab -e

*/2 * * * * /usr/sbin/logrotate -d /etc/logrotate.conf >> ~/Desktop/logfile.log 2>&1

e. Write a bash shell script that removes all the log files under the directory /var/log that have not been accessed in a week.

Open

removelog.sh
~/Desktop

```bash
#!/bin/bash

# Define the log directory
LOG_DIR="/var/log"

# Define the number of days (logs not accessed for more than 7 days will be deleted)
DAYS=7

# Find and remove log files not accessed in the last 7 days
find "$LOG_DIR" -type f -name "*.log" -atime +$DAYS -exec rm -f {} \;

# Print completion message
echo "Log cleanup completed. Removed logs not accessed in the last $DAYS days."
```

Deletes old log files that have not been accessed for more than 7 days inside the /var/log folder.

2 `"LOG_DIR="/var/log`

- يعرّف متغير **LOG_DIR** ليحدد مسار المجلد الذي يحتوي على ملفات **log**.

3 `DAYS=7`

- يعرّف متغير **DAYS** ويحدد عدد الأيام (هنا 7 أيام) التي سيتم استخدامها لتحديد الملفات القديمة.

4 **الأمر** `find`:

Edit Copy

```
find "$LOG_DIR" -type f -name "*.log" -atime +$DAYS -exec rm -f {} \;
```

- `"find "$LOG_DIR` → يبحث داخل مجلد **/var/log**.
- `type f-` → يضمن البحث عن **الملفات فقط** وليس المجلدات.
- `"name "*.log-` → يحدد البحث ليشمل فقط الملفات المنتهية بامتداد **.log**.
- `atime +$DAYS-` → يحدد الملفات التي لم يتم الوصول إليها منذ **أكثر من 7 أيام**.
- `;\ {} exec rm -f-` → ينفذ أمر الحذف `rm -f` لكل ملف تم العثور عليه.

3.4 Syslog messages

a. Show the basic information related to syslog daemon running on your machine.

```
cat /var/log/syslog
ps aux | grep rsyslog                # عرض العمليات المتعلقة بـ rsyslog
ls /etc/rsyslog.conf                 # التحقق من ملف الإعدادات الرئيسي
cat /etc/rsyslog.conf                # عرض إعدادات rsyslog
sudo netstat -tulnp | grep rsyslog   # عرض المنافذ التي يستمع إليها rsyslog
```

sudo tail -f /var/log/syslog # في الوقت الفعلي syslog مراقبة رسائل
sudo service rsyslog restart # إعادة تشغيل الخدمة

sudo gedit /etc/rsyslog.conf

sudo service rsyslog status

logger -p local1.notice “test1”

——→ in gedit /etc/rsyslog.conf write (local1.notice /home/ali/Desktop/file1.txt)

b. Write a bash shell script that generates a syslog message if your machine has been pinged.

The First Terminal : Execute the Script (syslog.sh)

syslog.sh ~/Desktop

```
#!/bin/bash


echo "Monitoring for incoming ping requests (including loopback). Press Ctrl+C to stop."


sudo tcpdump -i any icmp and icmp[icmptype] = icmp-echo -l | while read line
do

    logger "Ping detected: $line"
done
```

We use tcpdump to capture all ICMP echo request (ping) traffic, regardless of destination.

And we log the message to (syslog) when a ping is detected → logger “ping detected: $line”

On the Second Terminal:

ping 127.0.0.1

On the Third Terminal:

tail -f /var/log/syslog

We use `tail -f /var/log/syslog` to **continuously monitor system logs in real-time** and see new log entries as they are generated.

c. Generate a syslog message by your machine whose destination is your neighboring machine and make sure that this message has been captured by the neighboring machine.

First we make bridges in two pc's (PC1 and PC2) and restart them, to take diffrent IP.

Notes:

pc1 IP: 172.16.107.34

pc2 IP: 172.16.107.33

then,

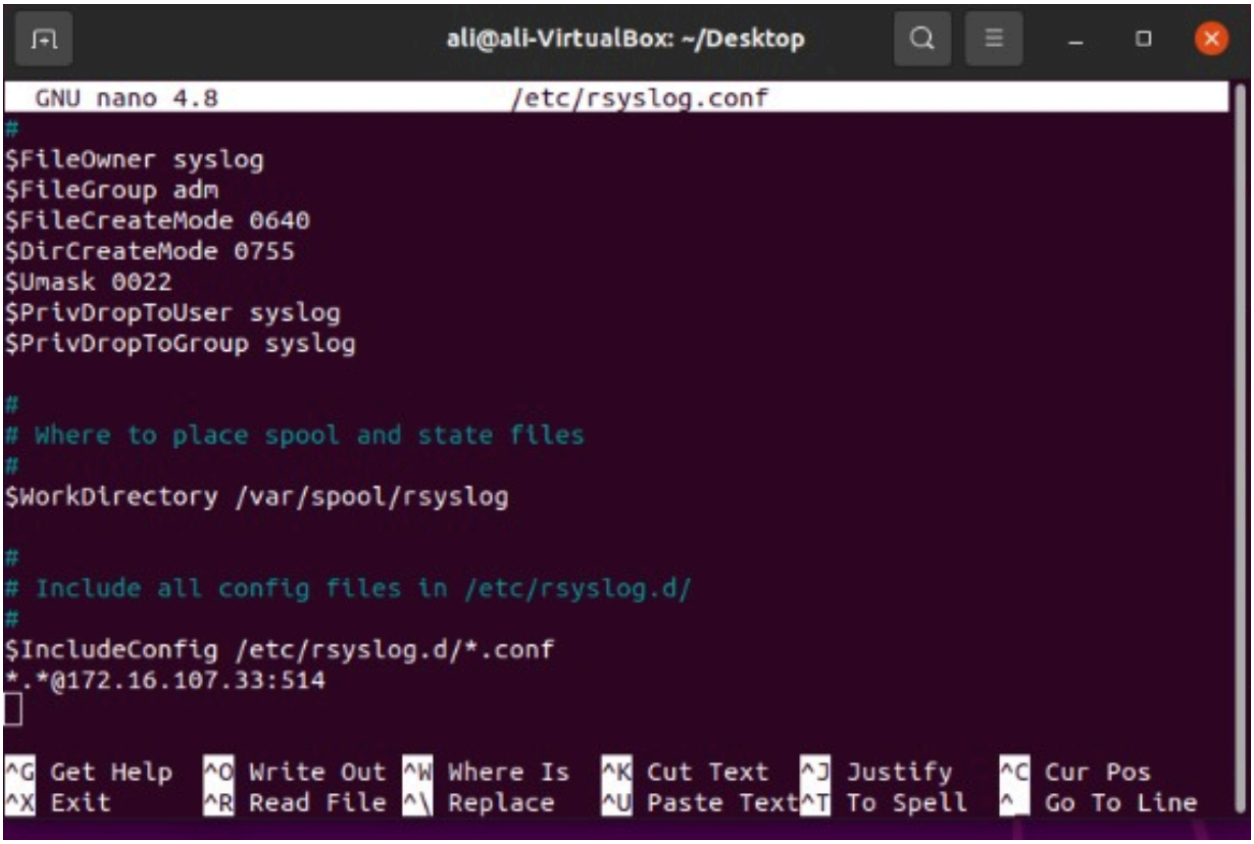
```
In PC1 open the rsyslog configuration file:
sudo nano /etc/rsyslog.conf

Add the Following Line to Forward Logs:
*.*@172.16.107.33:514


This line in the rsyslog configuration file is used to forward all log messages fr
om PC1 to another machine (PC2) at IP 172.16.107.33 using UDP port 514.

*.* → Matches all log messages (all priorities and all facilities).
@172.16.107.33:514 → Sends the logs to IP 172.16.107.33 (PC2) on port 514 usin
g UDP.
بشكل مختصر:
```

This configuration ensures that PC1 forwards all its logs to PC2,
which must be configured to receive syslog messages on UDP port 514.



To receive the syslog messages, the neighboring machine (PC2) must be conf igured to
accept incoming syslog messages.

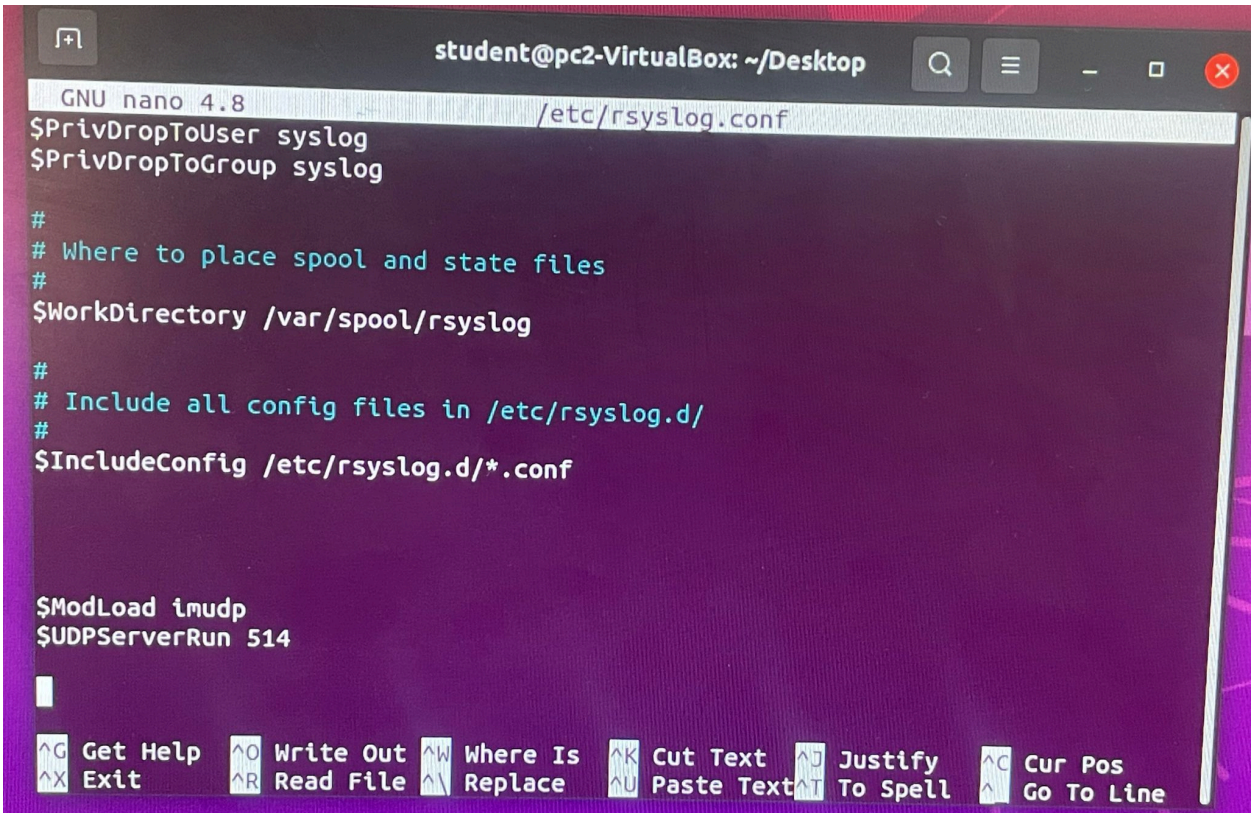
In PC2 open the rsyslog configuration file:
sudo nano /etc/rsyslog.conf

We added this:
$ModLoad imudp

$UDPServerRun 514

$ModLoad imudp → Enables receiving syslog messages over UDP in rsyslog.

$UDPServerRun 514 → Starts the syslog server to listen on UDP port 514 to re ceive
syslog messages from other devices.



Restart after editing /etc/rsyslog.conf.

sudo service rsyslog restart

You can generate a syslog message from your local machine (PC1) using logg er command.
Run the following command to send a log message to (PC2)

```
logger -n 172.16.107.33 -P 514 "Test syslog message from 172.16.107.34"
```

This will generate a syslog message that will be forwarded to 172.16.107.33

-P 514 → يحدد المنفذ الذي سيتم إرسال الرسالة إليه (514).

Verify the Syslog Message on PC2

on the neighboring machine (PC2), you can monitor the /var/log/syslog file to see if the message has been received. Run the following command to monit or the syslog in real-time:

```
sudo tail -f /var/log/syslog
```

Students: Ali Khuffash & Wajd Abd Al-Hadee
Done.