

Lab Objectives:

- create script to list all files in `/var/log` every 5 seconds and append the list to `/tmp/list.log` file

`touch script.sh`

`nano script.sh`






```
#!/bin/bash
while true
do
    ls /var/log >> /tmp/list.log
    sleep 5
done
```

`cd / tmp`




`touch list.log`

`cd ~`

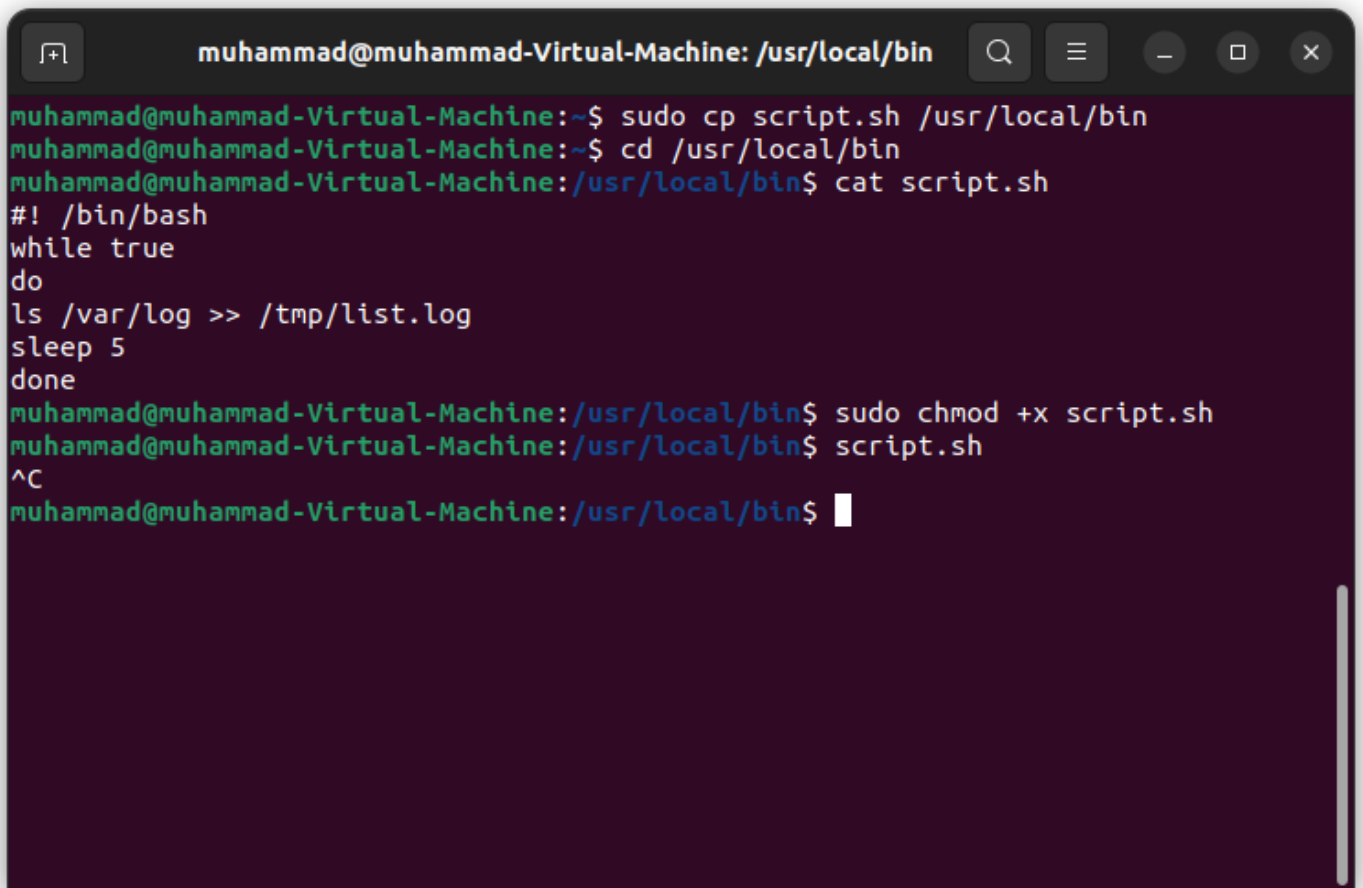
`bash script.sh`

Open list.log /tmp Save    

```
1 alternatives.log
2 apt
3 auth.log
4 boot.log
5 boot.log.1
6 boot.log.2
7 bootstrap.log
8 btmp
9 cups
10 dist-upgrade
11 dmesg
12 dmesg.0
13 dmesg.1.gz
14 dmesg.2.gz
15 dmesg.3.gz
16 dpkg.log
17 faillog
18 fontconfig.log
19 gdm3
20 gpu-manager.log
21 hp
22 installer
23 journal
24 kern.log
25 lastlog
26 oem-config.log
27 openvpn
28 private
29 speech-dispatcher
30 sssd
31 syslog
32 ubuntu-advantage.log
33 ubuntu-advantage-timer.log
34 unattended-upgrades
35 wtmp
36 Xorg.0.log
37 Xorg.0.log.old
```

Plain Text  Tab Width: 8  Ln 1, Col 1  INS

- copy the script to /usr/local/bin directory and set the execution attribute to it

A terminal window titled 'muhammad@muhammad-Virtual-Machine: /usr/local/bin' with standard window controls. The terminal shows a user named muhammad performing several commands: copying 'script.sh' to '/usr/local/bin' using 'sudo cp', changing the directory to '/usr/local/bin' with 'cd', viewing the script's content with 'cat script.sh', setting permissions with 'sudo chmod +x script.sh', and attempting to run 'script.sh' which is interrupted by a Ctrl-C (^C). The script content, as shown, is a bash script that enters an infinite loop, listing the contents of '/var/log' into '/tmp/list.log' and sleeping for 5 seconds.

```
muhammad@muhammad-Virtual-Machine: /usr/local/bin
muhammad@muhammad-Virtual-Machine:~$ sudo cp script.sh /usr/local/bin
muhammad@muhammad-Virtual-Machine:~$ cd /usr/local/bin
muhammad@muhammad-Virtual-Machine:/usr/local/bin$ cat script.sh
#!/bin/bash
while true
do
ls /var/log >> /tmp/list.log
sleep 5
done
muhammad@muhammad-Virtual-Machine:/usr/local/bin$ sudo chmod +x script.sh
muhammad@muhammad-Virtual-Machine:/usr/local/bin$ script.sh
^C
muhammad@muhammad-Virtual-Machine:/usr/local/bin$
```

- create systemd service file to execute the script as background service

```
muhammad@muhammad-Virtual-Machine: /etc/systemd/sys...
muhammad@muhammad-Virtual-Machine:~$
muhammad@muhammad-Virtual-Machine:~$ nano myServ.service
muhammad@muhammad-Virtual-Machine:~$ sudo systemctl daemon.reload
Unknown command verb daemon.reload.
muhammad@muhammad-Virtual-Machine:~$ sudo cp myServ.service /etc/systemd/system/
muhammad@muhammad-Virtual-Machine:~$ cd /etc/systemd/system
muhammad@muhammad-Virtual-Machine:/etc/systemd/system$ sudo systemctl daemon-reload
muhammad@muhammad-Virtual-Machine:/etc/systemd/system$ sudo service myServ start
muhammad@muhammad-Virtual-Machine:/etc/systemd/system$ sudo service myServ status
● myServ.service
   Loaded: loaded (/etc/systemd/system/myServ.service; disabled; vendor preset: enabled)
   Active: active (running) since Sat 2023-02-25 09:00:53 EST; 21s ago
     Main PID: 29635 (script.sh)
        Tasks: 2 (limit: 3371)
       Memory: 536.0K
          CPU: 24ms
       CGroup: /system.slice/myServ.service
               └─29635 /bin/bash /usr/local/bin/script.sh
                 └─29645 sleep 5

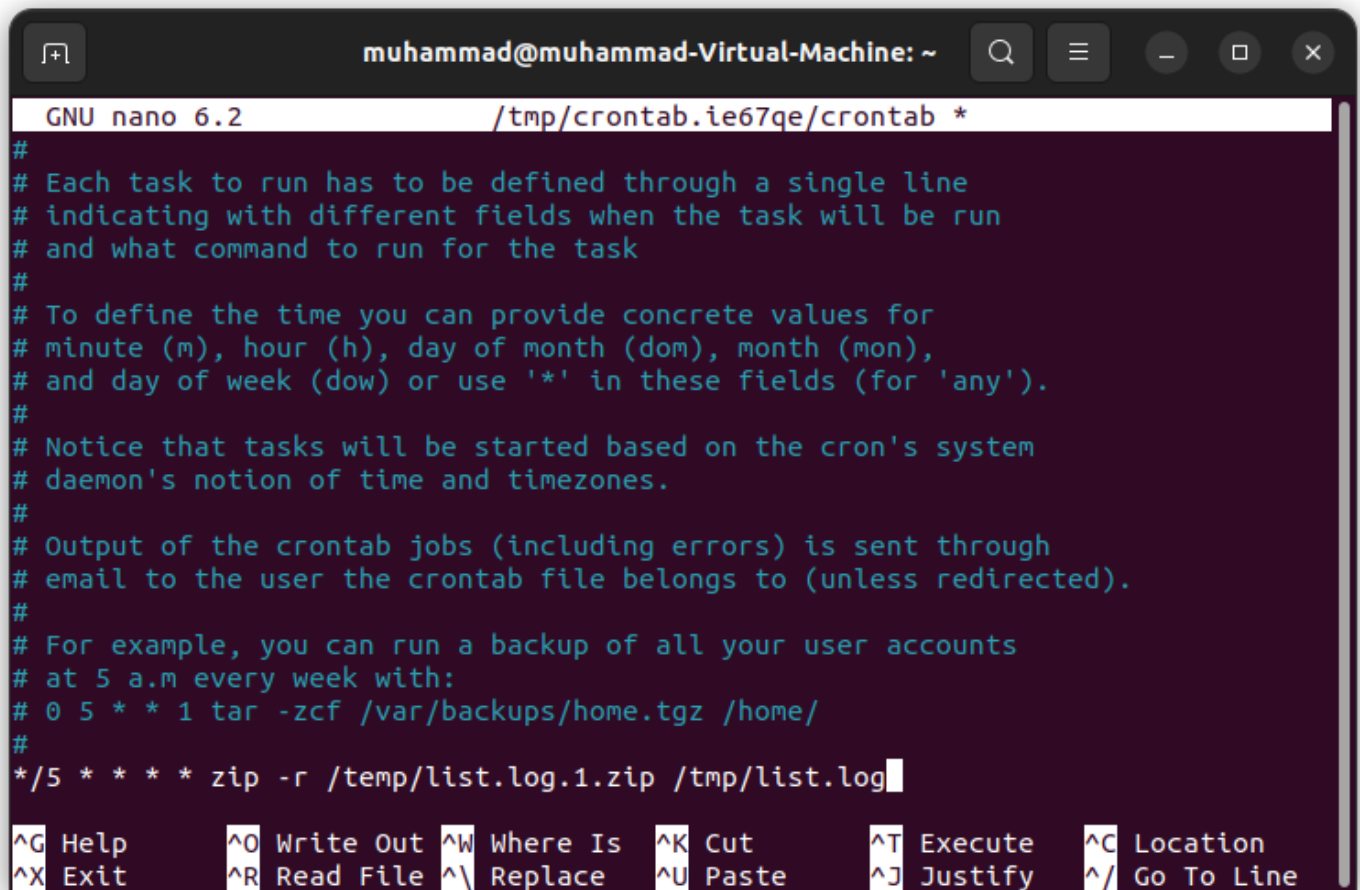
Feb 25 09:00:53 muhammad-Virtual-Machine systemd[1]: Started myServ.service.
Feb 25 09:00:53 muhammad-Virtual-Machine script.sh[29636]: /usr/local/bin/script.sh
```

- using `cron`, schedule a job every 5 minutes to copy `/tmp/list.log` file to `/tmp/list.log.1` and compress the file using `zip` command

`sudo apt-get install cron`

`crontab -e`

`* /5 * * * * zip -r /temp/list.log.1.zip /temp/list.log`



```
muhammad@muhammad-Virtual-Machine: ~
GNU nano 6.2 /tmp/crontab.ie67qe/crontab *
#
# 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
# 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/
#
*/5 * * * * zip -r /temp/list.log.1.zip /temp/list.log
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```