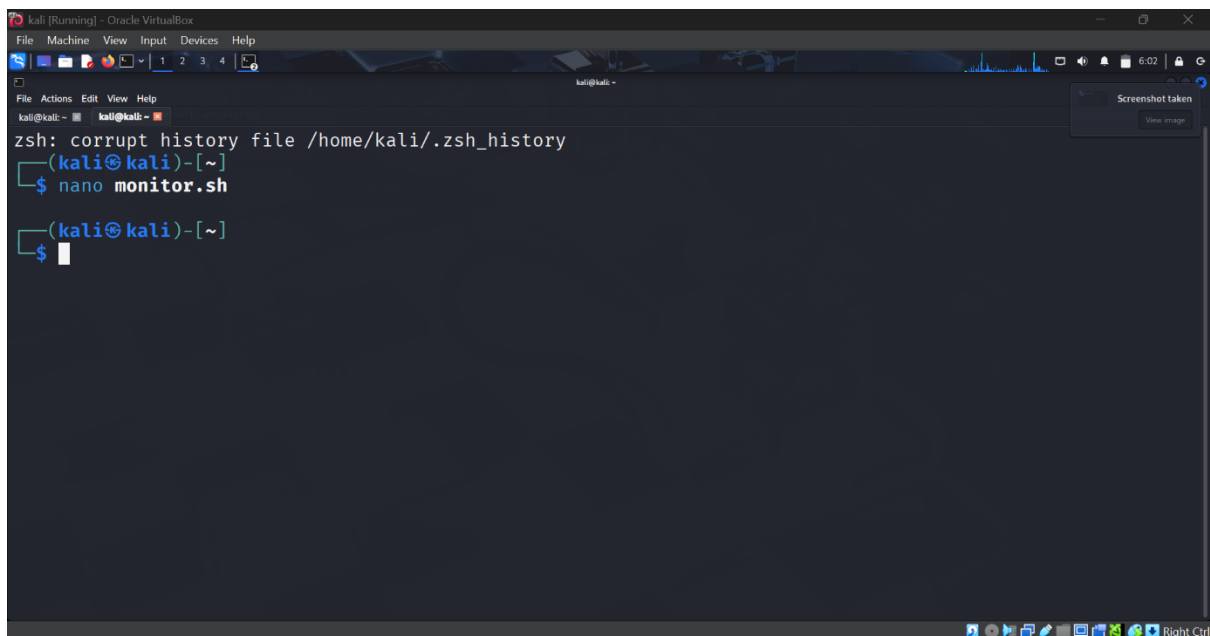TASK 3

Create a script 'monitor.sh' that:

- Checks if 'nginx' is running and starts it if not.

- Displays memory usage, CPU load, and disk usage.

- Logs all activities with timestamps.

Schedule it via cron to run every 5 minutes.

Step 1 :

Create a new script file

Step 2 :

Paste the code in the nano file.

```bash
#!/bin/bash

LOGFILE="/var/log/server_monitor.log"
TIMESTAMP=$(date "+%Y-%m-%d %H:%M:%S")

# Log start
echo "[$TIMESTAMP] Monitoring started" >> $LOGFILE

# Check if nginx is running
if ! pgrep nginx > /dev/null; then
    echo "[$TIMESTAMP] nginx is not running. Starting nginx ... " >> $LOGFILE
    systemctl start nginx
    if [ $? -eq 0 ]; then
        echo "[$TIMESTAMP] nginx started successfully." >> $LOGFILE
    else
        echo "[$TIMESTAMP] Failed to start nginx." >> $LOGFILE
    fi
else
    echo "[$TIMESTAMP] nginx is running." >> $LOGFILE
fi

# Memory usage
MEMORY=$(free -h | grep Mem | awk '{print "Used: "$3", Free: "$4}')
echo "[$TIMESTAMP] Memory Usage: $MEMORY" >> $LOGFILE

# CPU Load
CPU_LOAD=$(top -bn1 | grep "load average" | awk '{print $10 $11 $12}')
echo "[$TIMESTAMP] CPU Load: $CPU_LOAD" >> $LOGFILE

# Disk usage=DISK=$(df -h / | awk 'NR==2 {print "Used: "$3", Available: "$4}')
echo "[$TIMESTAMP] Disk Usage: $DISK" >> $LOGFILE

# Log end
echo "[$TIMESTAMP] Monitoring completed" >> $LOGFILE
```

Step 3 :

Make the script executable


```
-$ chmod +x /home/threat/monitor.sh
```

Step 4 :

Verify the output

```
└$ sudo /home/threat/monitor.sh
cat /home/threat/monitor.log

[2025-07-27 16:52:47] Monitoring started...
[2025-07-27 16:52:47] nginx is running.
[2025-07-27 16:52:47] Memory Usage:
               total        used        free      shared  buff/cache   available
Mem:           7.5Gi       3.9Gi       1.8Gi       785Mi       2.9Gi       3.6Gi
Swap:          7.7Gi          0B       7.7Gi
[2025-07-27 16:52:47] CPU Load:
 16:52:47 up  7:34,  1 user,  load average: 0.53, 0.57, 0.54
[2025-07-27 16:52:47] Disk Usage:
Filesystem      Size  Used Avail Use% Mounted on
udev            3.7G     0  3.7G   0% /dev
tmpfs           769M  2.1M  767M   1% /run
/dev/nvme0n1p2  460G   20G  418G   5% /
tmpfs           3.8G  9.7M  3.8G   1% /dev/shm
efivarfs        192K  120K   68K  64% /sys/firmware/efi/efivars
tmpfs           5.0M     0  5.0M   0% /run/lock
tmpfs           1.0M     0  1.0M   0% /run/credentials/systemd-journald.service
tmpfs           3.8G  3.9M  3.8G   1% /tmp
/dev/nvme0n1p1  975M  156K  974M   1% /boot/efi
tmpfs           769M  176K  769M   1% /run/user/1000
[2025-07-27 16:52:47] Monitoring completed.
```

Step 5 :

Automate with cron

```
┌──(threat@threat)-[~]
└$ crontab -e

no crontab for threat - using an empty one
Select an editor.  To change later, run select-editor again.
  1. /bin/nano        <---- easiest
  2. /usr/bin/vim.basic
  3. /usr/bin/vim.tiny

Choose 1-3 [1]: 1
crontab: installing new crontab
```