Bash Scripting

Project Title: Automation & Monitoring of Server Resources

By: Muazam Ali

07 Aug 2024

Q1: You will create a shell script for this project to automate the management and monitoring of server resources. You will be given ascenario in which you must write a script to carry out various server resource management and monitoring duties.

```
#!/bin/bash
```

This line indicates that the script is written in bash shell.

```
logs() {
   echo "$(date) - $1" >> server_monitor.log
   cat server_monitor.log
   cat alerts.log
}
```

The log_message function is used to save logs in server_monitor.log files. It takes ($1) one argument and appends it with the date $(date). Also the cat command is used to display the logs file content on the screen.

```
Disk_Usage() {
   echo "Inside Disk_Usage"
   disk_usage=$(df -h / | awk 'NR==2 {print $5}' | sed 's/%//')
   threshold=30
   if [ $disk_usage -gt $threshold ]; then
      logs "Disk usage is $disk_usage%, exceeds threshold of $threshold%"
      echo -e "\e[1;31mDisk usage is $disk_usage%, exceeds threshold of $threshold%. Sending alert...\e[0m" >> alerts.log
   else
      logs "Disk usage is $disk_usage%, within threshold"
   fi   }
```

The check_disk_usage() function is used to check the disk status and save its logs in a log file. The **df -h** command is used to display the disk space in human readable form like KB, MB etc.

The **awk ‘NR==2’{print $5}** is used to extract the fifth column of the second line from the output generated by df command.

The **sed 's/%//'** command is used to remove the percentage sign from extracted value. In the **if-conditional** statement the bash color output to differentiates alerts taken from *stackoverflow*. The **echo -e** command is used to print the results at screen with escape sequence characters.

```
Cpu_Usage() {
    echo "Inside Cpu_Usage"
    cpu_usage=$(top -bn1 | grep "Cpu(s)" | sed "s/.*, *\([0-9.]*\)%* id.*/\1/" | awk '{print 100 - $1}')
    threshold=30
    if [ $cpu_usage -gt $threshold ]; then
        logs "CPU usage is $cpu_usage%, exceeds threshold of $threshold%"
        echo -e "\e[1;32mCPU usage is $cpu_usage%, exceeds threshold of $threshold%. Sending alert...\e[0m" >> alerts.log
    else
        logs "CPU usage is $cpu_usage%, within threshold"
    fi
}
```

The check_cpu_usage() function stores and displays the logs of cpu usage status in the logs file.

The **top -bn1** command is used to display the CPU usage information once and then exit.

The **grep** filters the output of top which containing only **Cpu(s)** line.

This ***sed "s/.*, *\([0-9.]*\)%* id.*/\1/"*** command is used to extract the actual idle percentage of the cpu and remove everything.
The ***awk '{print 100 - $1}'*** Command is used to subtract the actual number by 100 and get the CPU usage state.

```
Memory_Usage() {
    echo "Inside Memory_Usage"
    memory_free=$(free | awk '/Mem/{print $4}')
    total_memory=$(free | awk '/Mem/{print $2}')
    threshold=$((total_memory / 10)) # 10% free memory threshold

    if [ $memory_free -lt $threshold ]; then
        logs "Available memory is low: $memory_free KB, falls below threshold of $threshold KB"
        echo -e "\e[1;33mAvailable memory is low: $memory_free KB, falls below threshold of $threshold KB. Sending alert...\e[0m" >> alerts.log
    else
        logs "Available memory is $memory_free KB, above threshold"
    fi
}
```

The check_memory_usage() function is used to check the memory usage of the server. It obtains this information using **free** command, calculates the available memory and compares it with threshold and generates logs accordingly.

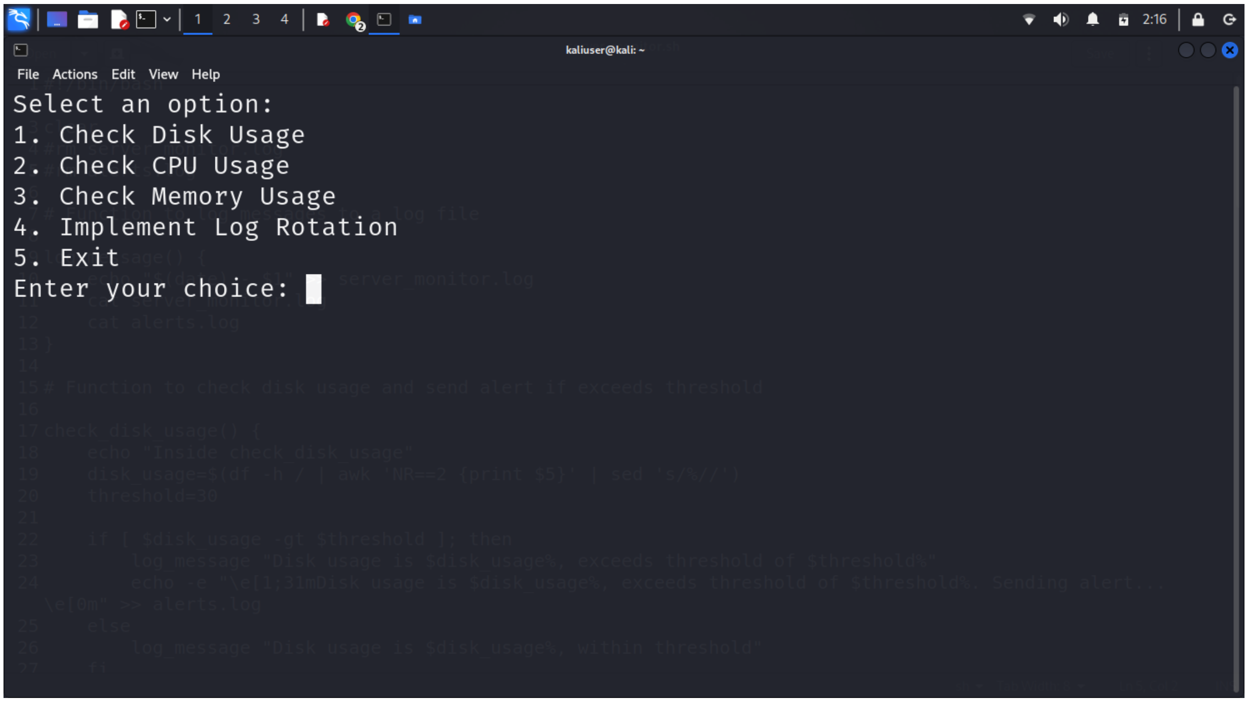
```
logfile_rotation() {
    log_file="/path/to/logfile.log"
    max_size="10M"
    if [ -f "$log_file" ]; then
        if [ $(stat -c %s "$log_file") -gt $(numfmt --from=auto "$max_size") ]; then
            logs "Rotating log file $log_file"
            mv "$log_file" "$log_file.$(date +%Y%m%d%H%M%S)"
            touch "$log_file"
            logrotate -vf /etc/logrotate.conf
        fi
    else
        logs "Log file $log_file not found"
    fi
}
```

This function checks if the log file exists and its size exceeds the maximum limit, it rotates the log file by renaming it and creates a new file by using **logrotate** command.
stat command is used to get the file size and compress it, **numfmt** command is used to convert the log file into bytes.

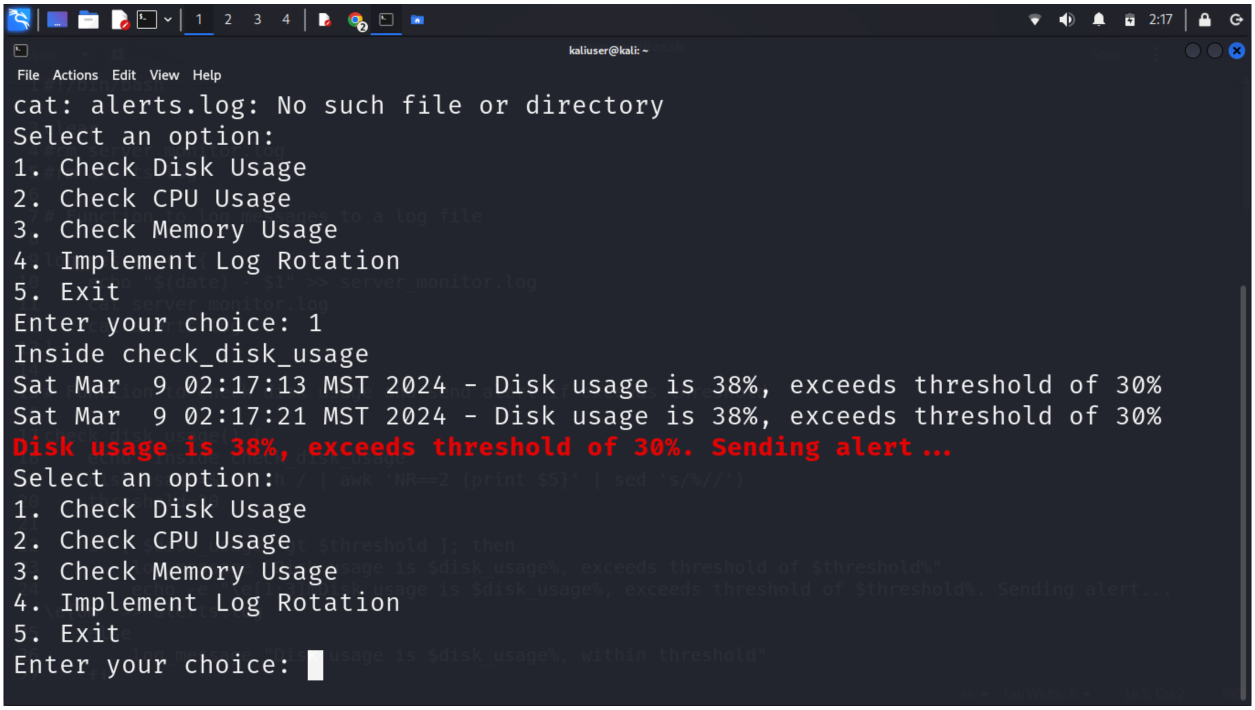
```
while true; do
    echo "Select an option:"
    echo "1. Check Disk Usage"
    echo "2. Check CPU Usage"
    echo "3. Check Memory Usage"
    echo "4. Implement Log Rotation"
    echo "5. Exit"
    read -p "Enter your choice: " choice
    case $choice in
        1)
            Disk_Usage
            ;;
        2)
            Cpu_Usage
            ;;
        3)
            Memory_Usage
            ;;
        4)
            logfile_rotation
            ;;
        5)
            echo "Exiting..."
            exit 0
            ;;
        *)
            echo "Invalid choice. Please select a valid option."
            ;;
    esac
done
```

This loop is used to display the main menu after execution of a file and display the output of these functions accordingly.

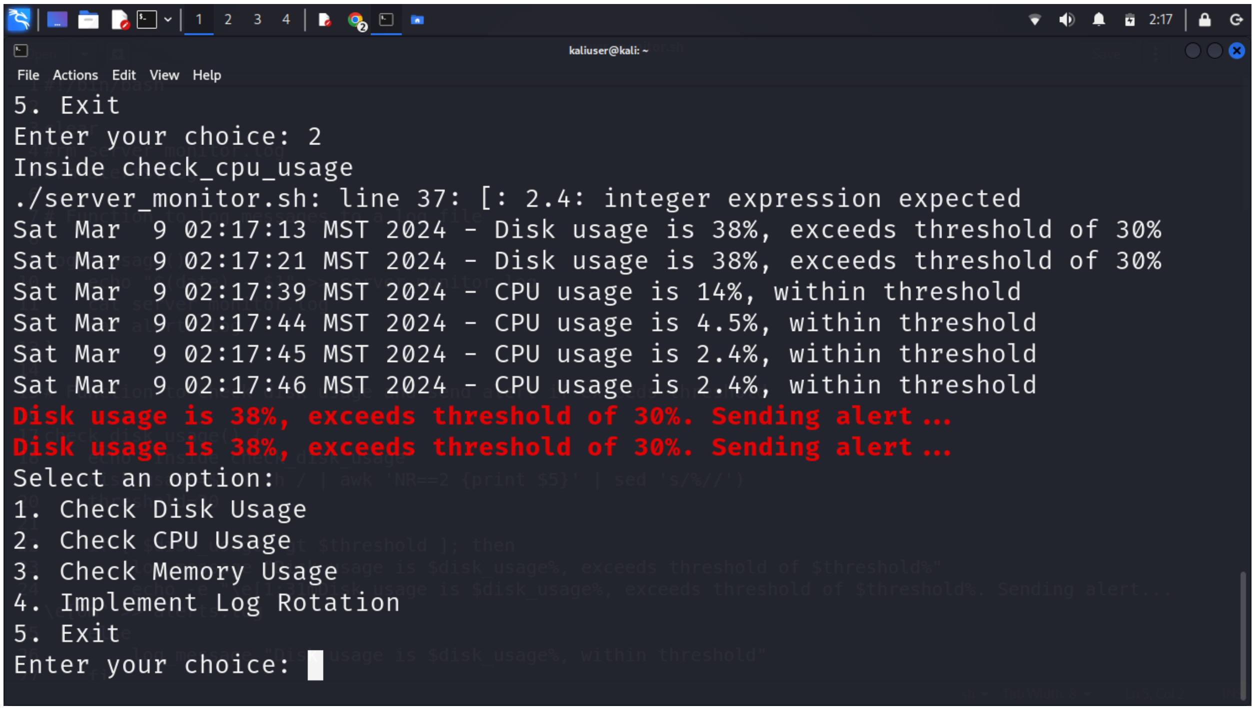
Here, some screenshots below to displays output of above bash file.



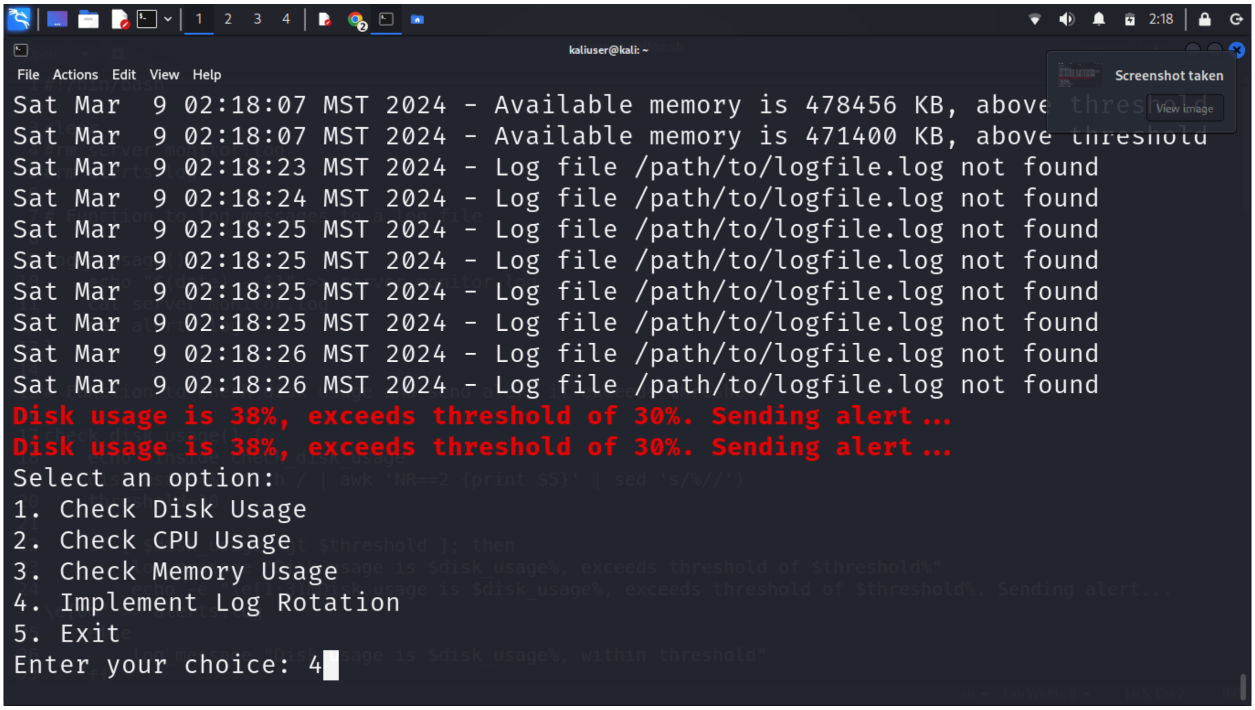
```
Select an option:
1. Check Disk Usage
2. Check CPU Usage
3. Check Memory Usage
4. Implement Log Rotation
5. Exit
Enter your choice: 
```



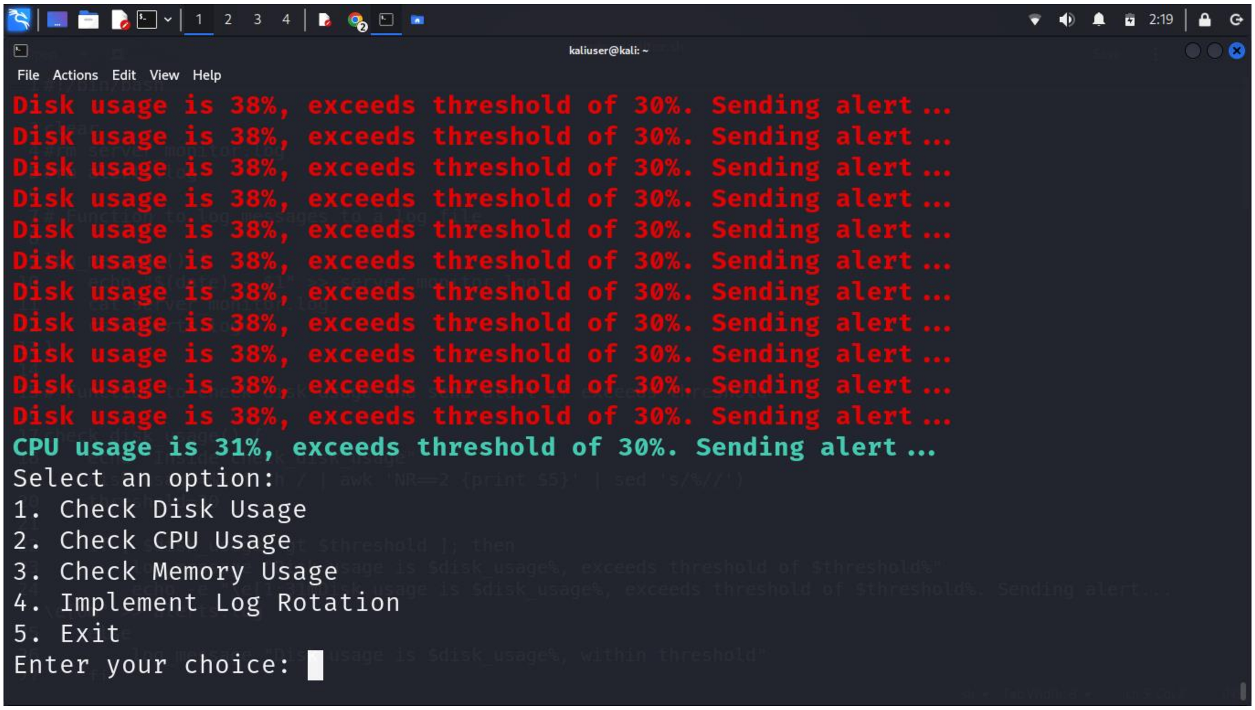
```
cat: alerts.log: No such file or directory
Select an option:
1. Check Disk Usage
2. Check CPU Usage
3. Check Memory Usage
4. Implement Log Rotation
5. Exit
Enter your choice: 1
Inside check_disk_usage
Sat Mar  9 02:17:13 MST 2024 - Disk usage is 38%, exceeds threshold of 30%
Sat Mar  9 02:17:21 MST 2024 - Disk usage is 38%, exceeds threshold of 30%
Disk usage is 38%, exceeds threshold of 30%. Sending alert...
Select an option:
1. Check Disk Usage
2. Check CPU Usage
3. Check Memory Usage
4. Implement Log Rotation
5. Exit
Enter your choice: 
```



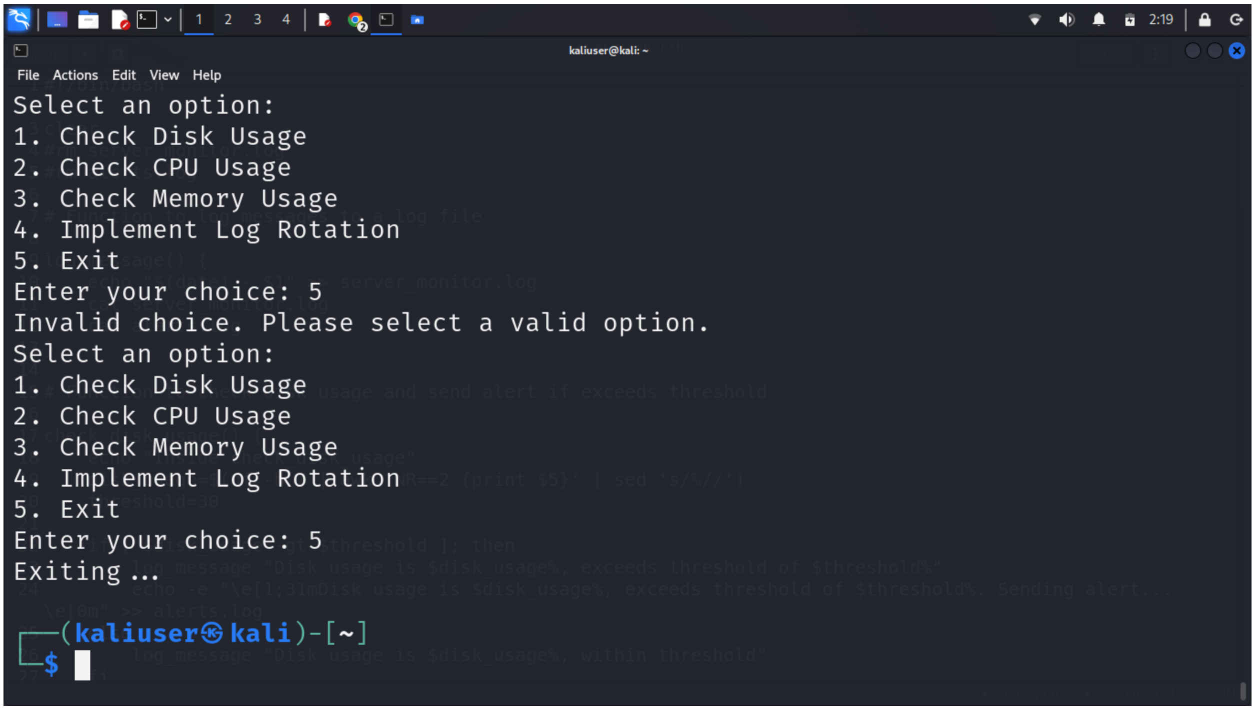
```
5. Exit
Enter your choice: 2
Inside check_cpu_usage
./server_monitor.sh: line 37: [: 2.4: integer expression expected
Sat Mar  9 02:17:13 MST 2024 - Disk usage is 38%, exceeds threshold of 30%
Sat Mar  9 02:17:21 MST 2024 - Disk usage is 38%, exceeds threshold of 30%
Sat Mar  9 02:17:39 MST 2024 - CPU usage is 14%, within threshold
Sat Mar  9 02:17:44 MST 2024 - CPU usage is 4.5%, within threshold
Sat Mar  9 02:17:45 MST 2024 - CPU usage is 2.4%, within threshold
Sat Mar  9 02:17:46 MST 2024 - CPU usage is 2.4%, within threshold
Disk usage is 38%, exceeds threshold of 30%. Sending alert...
Disk usage is 38%, exceeds threshold of 30%. Sending alert...
Select an option:
1. Check Disk Usage
2. Check CPU Usage
3. Check Memory Usage
4. Implement Log Rotation
5. Exit
Enter your choice: 
```



```
Sat Mar  9 02:18:07 MST 2024 - Available memory is 478456 KB, above threshold
Sat Mar  9 02:18:07 MST 2024 - Available memory is 471400 KB, above threshold
Sat Mar  9 02:18:23 MST 2024 - Log file /path/to/logfile.log not found
Sat Mar  9 02:18:24 MST 2024 - Log file /path/to/logfile.log not found
Sat Mar  9 02:18:25 MST 2024 - Log file /path/to/logfile.log not found
Sat Mar  9 02:18:25 MST 2024 - Log file /path/to/logfile.log not found
Sat Mar  9 02:18:25 MST 2024 - Log file /path/to/logfile.log not found
Sat Mar  9 02:18:25 MST 2024 - Log file /path/to/logfile.log not found
Sat Mar  9 02:18:26 MST 2024 - Log file /path/to/logfile.log not found
Sat Mar  9 02:18:26 MST 2024 - Log file /path/to/logfile.log not found
Disk usage is 38%, exceeds threshold of 30%. Sending alert...
Disk usage is 38%, exceeds threshold of 30%. Sending alert...
Select an option:
1. Check Disk Usage
2. Check CPU Usage
3. Check Memory Usage
4. Implement Log Rotation
5. Exit
Enter your choice: 4
```



```
Disk usage is 38%, exceeds threshold of 30%. Sending alert ...
Disk usage is 38%, exceeds threshold of 30%. Sending alert ...
Disk usage is 38%, exceeds threshold of 30%. Sending alert ...
Disk usage is 38%, exceeds threshold of 30%. Sending alert ...
Disk usage is 38%, exceeds threshold of 30%. Sending alert ...
Disk usage is 38%, exceeds threshold of 30%. Sending alert ...
Disk usage is 38%, exceeds threshold of 30%. Sending alert ...
Disk usage is 38%, exceeds threshold of 30%. Sending alert ...
Disk usage is 38%, exceeds threshold of 30%. Sending alert ...
Disk usage is 38%, exceeds threshold of 30%. Sending alert ...
Disk usage is 38%, exceeds threshold of 30%. Sending alert ...
CPU usage is 31%, exceeds threshold of 30%. Sending alert ...
Select an option:
1. Check Disk Usage
2. Check CPU Usage
3. Check Memory Usage
4. Implement Log Rotation
5. Exit
Enter your choice:
```



```
Select an option:
1. Check Disk Usage
2. Check CPU Usage
3. Check Memory Usage
4. Implement Log Rotation
5. Exit
Enter your choice: 5
Invalid choice. Please select a valid option.
Select an option:
1. Check Disk Usage
2. Check CPU Usage
3. Check Memory Usage
4. Implement Log Rotation
5. Exit
Enter your choice: 5
Exiting ...

┌──(kaliuser㉿kali)-[~]
└─$
```