

PSMonitor Script Documentation

Abraham Reines

February 22, 2024

1 Script Requirements

The `psmonitor.sh` script scans the system process table; it displays the process running on the system. A specific amount of intervals for a specified number of iterations is achieved. It takes two command-line arguments (CLAs) to set the time between scans and the number of scans.

2 Usage

The script is invoked with the following syntax:

```
1 ./psmonitor.sh [-t tseconds] [-n count]
```

where `-t tseconds` specifies the time interval between each scan, and `-n count` specifies the number of times the scan is performed.

3 Error Checking

The script includes error checking for invalid options, missing argument values, and non-numeric input for time intervals and count.

4 Handling Interrupts

Interrupts, such as Ctrl-C, are gracefully handled by the script to provide a user-friendly exit message.

5 Script Listing

```
1 #!/bin/bash
2 # Author: Abraham Reines
3 # Date: 14-02-2024 09:35:07
4
5 tseconds=1
6 count=5
7
8 # Function for usage
9 show_usage() {
10     echo "Usage: $0 [-t tseconds] [-n count]"
11     exit 1
12 }
13
14 # Function to handle interrupts
15 handle_interrupt() {
16     echo "Interruption occurred. Exiting with grace..."
17     exit 2
18 }
19
20 # SIGINT Trap (Ctrl-C)
21 trap handle_interrupt SIGINT
```

```

22
23 # Parsing
24 while getopts ":t:n:" opt; do
25     case ${opt} in
26         t )
27             tseconds=$OPTARG
28             ;;
29         n )
30             count=$OPTARG
31             ;;
32         \? )
33             echo "Invalid Option: -$OPTARG" 1>&2
34             show_usage
35             ;;
36         : )
37             echo "Option -$OPTARG requires an argument." 1>&2
38             show_usage
39             ;;
40     esac
41 done
42
43 # Check for integers
44 if ! [[ $tseconds =~ ^[0-9]+$ ]] || ! [[ $count =~ ^[0-9]+$ ]]; then
45     echo "Error: tseconds and count must be positive and integers."
46     exit 3
47 fi
48
49 # Main loop
50 for (( i=0; i<$count; i++ )); do
51     echo $(date)
52     ps -ef
53     sleep $tseconds
54 done
55
56 echo
57 echo
58 echo "Program is written by Abraham Reines. This work complies with the JMU
    ↪ honor code. I did not give or receive unauthorized help on this
    ↪ assignment. Exiting..."

```

6 Sample Output Listing

For the first 10 lines:

```

1 Tue Feb 13 10:59:20 PST 2024
2  UID  PID  PPID  C STIME  TTY          TIME CMD
3    0    1    0    0  9:42AM ??          1:37.34 /sbin/launchd
4    0   329    1    0  9:43AM ??          0:35.02 /usr/libexec/logd
5    0   330    1    0  9:43AM ??          0:00.04 /usr/libexec/smd
6    0   331    1    0  9:43AM ??          0:01.45 /usr/libexec/UserEventAgent
    ↪ (System)
7    0   333    1    0  9:43AM ??          0:00.16 /System/Library/
    ↪ PrivateFrameworks/Uninstall.framework/Resources/uninstalld
8    0   334    1    0  9:43AM ??          0:15.89 /System/Library/Frameworks/
    ↪ CoreServices.framework/Versions/A/Frameworks/FSEvents.framework/
    ↪ Versions/A/Support/fseventsd
9    0   335    1    0  9:43AM ??          0:00.83 /System/Library/
    ↪ PrivateFrameworks/MediaRemote.framework/Support/mediaremoted
10   0   338    1    0  9:43AM ??          0:00.90 /usr/sbin/systemstats --
    ↪ daemon

```

For the last 10 lines:

```
1      0      856      671      0  9:43AM ttys001      0:00.01 login -pf abrahamreines
2    501      861      856      0  9:43AM ttys001      0:00.10 -zsh
3      0      860      671      0  9:43AM ttys002      0:00.01 login -pfl abrahamreines /
      ↪ bin/bash -c exec -la zsh /bin/zsh
4    501      864      860      0  9:43AM ttys002      0:00.10 -zsh
5      0      862      671      0  9:43AM ttys003      0:00.01 login -pfl abrahamreines /
      ↪ bin/bash -c exec -la zsh /bin/zsh
6    501      886      862      0  9:43AM ttys003      0:00.10 -zsh
7    501    8615    3718      0 10:55AM ttys009      0:00.12 /bin/zsh -il
8    501   10168    8615      0 10:59AM ttys009      0:00.02 /bin/bash ./psmonitor.sh -t
      ↪ 1 -n 10
9      0   10240   10168      0 10:59AM ttys009      0:00.01 ps -ef
10 Program is written by Abraham Reines. This work complies with the JMU honor
      ↪ code. I did not give or receive unauthorized help on this assignment.
      ↪ Exiting...
```

7 Executing the Script

To execute the script, first ensure it has the appropriate permissions set:

```
1 chmod +x psmonitor.sh
```

Then run the script by providing the desired arguments for time interval and count, e.g.:

```
1 ./psmonitor.sh -t 1 -n 10 > psmonitor_output.txt
```

or use the default values by not providing any arguments.

Academic Integrity Pledge

“This work complies with the JMU honor code. I did not give or receive unauthorized help on this assignment.”