Name: Mosab Fathy Ramadan Mohamed
Group: B20-SD-01
Lab 7: Processes and signals

1. What are zombie processes? How can you find and kill them?

Answer:

A zombie process is a process that has completed execution but still has an entry in the process table

To find them we use “top -b1 -n1 | grep Z”
To kill we use “kill -s SIGCHLD ppid”

2. What are the differences between `kill`, `killall`, and `pkill`?

Answer:

kill: Kill the process by specifying its PID

killall: Kill the processes by name, will end all processes that have a matching name

pkill: Send a signal to the process based on its name, you can send a signal to any process by specifying the full name or partial name. So there is no need for you to find out the PID of the process to send the signal

3. Run the `top` command on your system and annotate the data in the `Tasks` and `%Cpu(s)` lines of your output. Provide single sentence explanations for each of the data presented in these two lines.

Answer:

```
iviosab@iviosab:~$ top

top - 22:08:00 up 6 days, 14:36,  1 user,  load average: 0,80, 0,82, 0,64
Tasks: 395 total,   2 running, 393 sleeping,   0 stopped,   0 zombie
%Cpu(s):  2,8 us,  1,0 sy,  0,0 ni, 96,1 id,  0,1 wa,  0,0 hi,  0,0 si,  0,0 st
MiB Mem :  15907,6 total,    294,5 free,   8228,1 used,   7384,9 buff/cache
MiB Swap:   2048,0 total,   1899,2 free,    148,8 used.   6964,6 avail Mem
```

Tasks: 395 total processes in the system with 2 running, 393 sleeping for resources, 0 already stopped and 0 waiting for their parent to release.

CPU: 2.8% user process, 1.0 for the kernel, 0.0 for manual values, 96.1 the time idle of the CPU, 0.1 wait time.

4. Execute the following bash command:

```
$ bash -c "exec -a fun${RANDOM}process sleep infinity" &
```

   - Assume that there are multiple of such processes. To simulate this, you can run the command more than once.
   - Write a bash script that will locate and kill all the processes created by this command.
   - Display status messages when one of such processes is found, and when the process is killed. Additionally, display a message when the process is not found.
   - Your script should work on any machine it is executed on.
   - Be extremely careful and be as accurate as possible when finding this process. You don't want to kill the wrong process.

   > Show test results in the form of screenshots.

Answer:

```
#!/bin/bash
kill $(ps aux | grep -E "fun[0-9]+process" | awk '{print $2}')
~
```

```
iviosab@iviosab:~/Desktop/GitHub/f22-sna/Week-7$ bash -c "exec -a fun${RANDOM}process sl
eep infinity" &
[1] 1048237
iviosab@iviosab:~/Desktop/GitHub/f22-sna/Week-7$ bash -c "exec -a fun${RANDOM}process sl
eep infinity" &
[2] 1048238
iviosab@iviosab:~/Desktop/GitHub/f22-sna/Week-7$ bash -c "exec -a fun${RANDOM}process sl
eep infinity" &
[3] 1048241
iviosab@iviosab:~/Desktop/GitHub/f22-sna/Week-7$ ps aux | grep -E "fun[0-9]+process"
iviosab  1048237  0.0  0.0   8372  1044 pts/0    S    21:54   0:00 fun8229process infini
ty
iviosab  1048238  0.0  0.0   8372  1008 pts/0    S    21:54   0:00 fun29153process infin
ity
iviosab  1048241  0.0  0.0   8372  1040 pts/0    S    21:54   0:00 fun2136process infini
ty
iviosab@iviosab:~/Desktop/GitHub/f22-sna/Week-7$ sudo bash ex4.sh
[1]   Terminated              bash -c "exec -a fun${RANDOM}process sleep infinity"
[2]-  Terminated              bash -c "exec -a fun${RANDOM}process sleep infinity"
[3]+  Terminated              bash -c "exec -a fun${RANDOM}process sleep infinity"
iviosab@iviosab:~/Desktop/GitHub/f22-sna/Week-7$
```

5. Write a bash script that loops infinitely and prints “Hello world!” every ten seconds. It should print “Interrupt received” when it receives `SIGUSR1`.
    - Show the script in the report, and show how you’re sending the signal to it.

Answer:

```
#!/bin/bash

function interrupt()
{
        echo "Interrupt received"
}

trap interrupt USR1

while :
do
        echo "Hello world!"
        sleep 10
done
```

```
iviosab@iviosab:~/Desktop/GitHub/f22-sna/Week-7$ bash ex5.sh
Hello world!
Hello world!
Interrupt received
Hello world!
```

```
iviosab@iviosab:~$ ps -ef | grep ex5
iviosab   999876  998789  0 22:18 pts/0    00:00:00 bash ex5.sh
iviosab   999893  999580  0 22:19 pts/1    00:00:00 grep --color=auto ex5
iviosab@iviosab:~$ sudo kill -SIGUSR1 999876
```

6. Write a bash script to monitor CPU usage, memory usage, and disk space usage.
    - For testing purposes, the check should execute every 15 seconds.
    - The usage statistics should be saved to a log file `/var/log/system_utilization.log`.
    - One line of log should contain the timestamp, the % of CPU in use, the % of memory in use, and the % of disk space used.
    - The log should contain descriptive information that will make it easy to understand.

    > Show log samples created by this service in your report.

Answer:

```
#!/bin/bash
while true
do
  echo "date: $(date)" >> /var/log/system_utilization.log
  echo "$(cat /proc/stat |grep cpu |tail -1|awk '{print ($5*100)/($2+$3+$4+$5+$6+$7+$8+$
9+$10) }'|awk '{print "CPU Usage: " 100-$1 "%"}')" >> /var/log/system_utilization.log
  echo "$(free | grep "Mem" | awk '{print "Memory usage: " ($3/$2)*100 "%"}')" >> /var/l
og/system_utilization.log
  echo "Disk usage: $(df / | awk '{print $5}'| tail -n 1)" >> /var/log/system_utilizatio
n.log
  echo "------------15 Seconds Interval------------" >> /var/log/system_utilization.log
  sleep 15
done
~
```

```
iviosab@iviosab:~/Desktop/GitHub/f22-sna/Week-7$ sudo bash ex6.sh
```

```
iviosab@iviosab:~/Desktop/GitHub/f22-sna/Week-7$ cat /var/log/system_utilization.log
date: Вс 16 окт 2022 21:48:43 MSK
CPU Usage: 44,3449%
Memory usage: 55,203%
Disk usage: 55%
------------15 Seconds Interval------------
iviosab@iviosab:~/Desktop/GitHub/f22-sna/Week-7$ cat /var/log/system_utilization.log
date: Вс 16 окт 2022 21:48:43 MSK
CPU Usage: 44,3449%
Memory usage: 55,203%
Disk usage: 55%
------------15 Seconds Interval------------
date: Вс 16 окт 2022 21:48:58 MSK
CPU Usage: 44,3358%
Memory usage: 55,1428%
Disk usage: 55%
------------15 Seconds Interval------------
iviosab@iviosab:~/Desktop/GitHub/f22-sna/Week-7$
```