RIPHAH INTERNATIONAL UNIVERSITY, ISLAMABAD

Lab 12

Bachelors of Computer science – 5th semester

Subject: Operating System Lab

Submitted to: Ma'am Kausar Nasreen Khattak

Submitted by: Tooba Baqai

Sap Id: 46489

Date: 10th November, 2024

Lab Tasks:

Q1: Which command would you use to find the process ID (PID) of a process named OSLab without running it. After obtaining the PID, which command would you use to kill the process?

Answer:

Command: To find the process ID (PID) of a process named OSLab, we use: **pidof OSLab**
This command returns the PID of process with the name OSLab.
After obtaining the PID, we can terminate the process with the kill command:
Command: kill <PID>
If we need to forcefully terminate the process, we can add the -9 option: **kill -9 <PID>**

Q2: How would you write a script that uses a signal trap to handle specific signals, and what is the purpose of a signal trap in such a script?

Answer:

Code:

```
#!/bin/bash

# Define a function to handle signals

handle_signal() {

  echo "Signal caught, performing cleanup..."

  # Add any cleanup or exit steps here

  exit 0

}

# Set up a trap to catch SIGINT (Ctrl+C) and SIGTERM

trap handle_signal SIGINT SIGTERM
```

Main script loop

while true; do

  echo "Running... Press Ctrl+C to stop."

  sleep 1

done

```
[root@localhost ~]# vi signal.sh
```

```
#!/bin/bash
# Define a function to handle signals
handle_signal() {
  echo "Signal caught, performing cleanup..."
  # Add any cleanup or exit steps here
  exit 0
}
# Set up a trap to catch SIGINT (Ctrl+C) and SIGTERM
trap handle_signal SIGINT SIGTERM
# Main script loop
while true; do
  echo "Running... Press Ctrl+C to stop."
  sleep 1
done
```

Esc

Shift+:

wq

Enter

```
[root@localhost ~]# chmod 777 signal.sh
[root@localhost ~]# ./signal.sh
Running... Press Ctrl+C to stop.
Running... Press Ctrl+C to stop.
Running... Press Ctrl+C to stop.
Running... Press Ctrl+C to stop.
Running... Press Ctrl+C to stop.
Running... Press Ctrl+C to stop.
^CSignal caught, performing cleanup...
```

Function Definition: handle_signal is defined to execute any cleanup tasks, and it prints a message before exiting the script with a status of 0 (indicating a normal exit).

Setting Up the Trap: trap handle_signal SIGINT SIGTERM tells the script to call the handle_signal function when it receives either a SIGINT (interrupt, often triggered by pressing Ctrl+C) or SIGTERM (termination signal).

Infinite Loop: The script runs an infinite loop, printing "Running... Press Ctrl+C to stop." every second until interrupted.

Purpose:

A signal trap allows a script to handle interruptions gracefully by executing specific commands when a signal is received. A signal trap allows a script to intercept signals like SIGINT (interrupt, typically from pressing Ctrl+C), SIGTERM (terminate), and others. This lets the script perform cleanup tasks, such as saving data, closing files, or releasing resources, before exiting gracefully.