Documentation 377 Lab 4

Gia Lee, Valerie So, William Robson

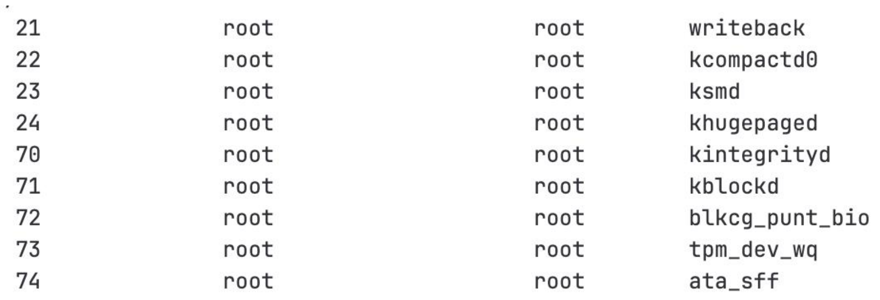
Purpose

The purpose of this lab is to use shell scripting to display and list all the running processes on the lab machines. This is a good demonstration of how an operating system understands and handles processes. It is also building an understanding of just how many processes exist and run simultaneously even on a basic system.

The shell takes input as flags to tailor what information the user wants to see, then iterates through each process and prints the requested information about the processes to the console.

Output

The shell prints information on each process (depending on which information was flagged on the input), to the console. There are many lines of output as there are many processes.



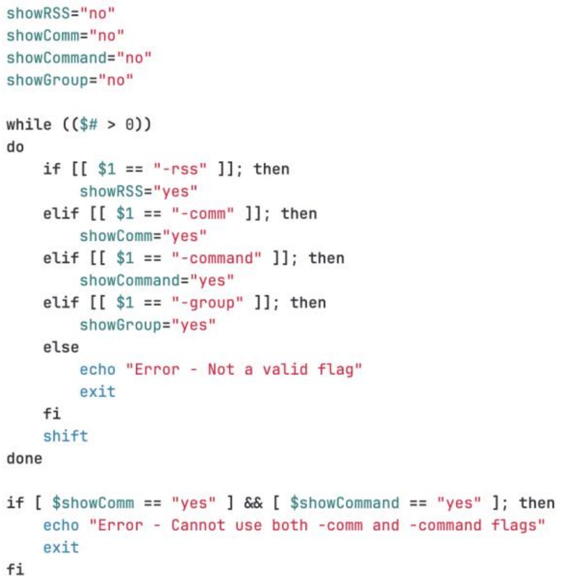
```
21              root                    root        writeback
22              root                    root        kcompactd0
23              root                    root        ksmd
24              root                    root        khugepaged
70              root                    root        kintegrityd
71              root                    root        kblockd
72              root                    root        blkcg_punt_bio
73              root                    root        tpm_dev_wq
74              root                    root        ata_sff
```

Flag Logic

Flags are "no" by default,

Arguments are checked, and then flags are set to "yes" if the data represented by that flag are wanted to be displayed.

There is also a check to make sure comm and command are not displayed at the same time, as they are mutually exclusive



```
showRSS="no"
showComm="no"
showCommand="no"
showGroup="no"

while (($# > 0))
do
    if [[ $1 == "-rss" ]]; then
        showRSS="yes"
    elif [[ $1 == "-comm" ]]; then
        showComm="yes"
    elif [[ $1 == "-command" ]]; then
        showCommand="yes"
    elif [[ $1 == "-group" ]]; then
        showGroup="yes"
    else
        echo "Error - Not a valid flag"
        exit
    fi
    shift
done

if [ $showComm == "yes" ] && [ $showCommand == "yes" ]; then
    echo "Error - Cannot use both -comm and -command flags"
    exit
fi
```

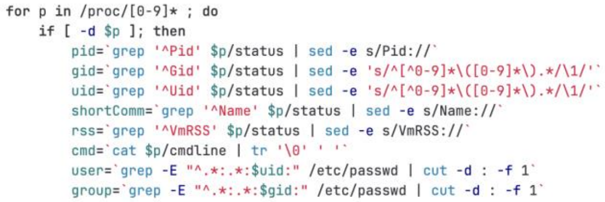
Data Retrieval

Retrieve data by using a for-loop, iterating over all directories for each process.

Within the for loop:

1. Extract information from status file for a process using “grep” and “sed”. Extracting Process ID, Group ID, User ID, RSS and the short command

2. Convert numeric user and group IDs to user and group variable by extracting information from /etc/passwd.



```
for p in /proc/[0-9]* ; do
    if [ -d $p ]; then
        pid=`grep '^Pid' $p/status | sed -e s/Pid://`
        gid=`grep '^Gid' $p/status | sed -e 's/^[^0-9]*\([0-9]*\).*/\1/'`
        uid=`grep '^Uid' $p/status | sed -e 's/^[^0-9]*\([0-9]*\).*/\1/'`
        shortComm=`grep '^Name' $p/status | sed -e s/Name://`
        rss=`grep '^VmRSS' $p/status | sed -e s/VmRSS://`
        cmd=`cat $p/cmdline | tr '\0' ' '`
        user=`grep -E "^.*:.*:$uid:" /etc/passwd | cut -d : -f 1`
        group=`grep -E "^.*:.*:$gid:" /etc/passwd | cut -d : -f 1`
```