OSYS2022 Linux Scripting
Assignment 2 – Basic Scripting
Issued: February 2, 2024

Introduction

The time has come for us to build our first legitimate script. One of the most common purposes for a script is for providing a report on specific criteria that need to be monitored such as the performance of a system, critical services on a server, or the network connectivity of various hosts. Let us use this type of script as a way for us to begin piecing together some of the things we have been learning so far.
In this assignment we will build out the skeleton of the script, including sending all output to a file and formatting the output to make it easier to read. In module 3 we will build upon this script by adding in more commands, more complexity, more logic, more EVERYTHING!

Required Resources

You will require the following to perform this assignment:
• VMWare Workstation Professional
• Linux VM of your choice (Recommend Debian-based distribution to mirror in-class content)

Marking

Outcome(s) being measured:
• Outcome 1: Implement management and configuration settings in a Linux environment
• Outcome 2: Apply command line syntax and skills required to manage a Linux system
• Outcome 3: Secure a Linux environment using scripting to manage users, groups, and process creation

Task 1 - Create a Script

You are to create a script that meets the following criteria:
• Stored in your ~/bin directory
• Name of script: osys2022_script
• Shebang is configured
• Permissions are configured so the owner of the script can execute it, with no permissions provided for anyone else whether a group or individual ○ Take **screenshot** showing the permissions for the file

• The $PATH variable lists the ~/bin directory (remember that the ~ icon is a short-form for a specific directory, so you will need to check the $PATH variable for both) ○ Take **screenshot** showing the directory is listed in the $PATH variable

With the script created, lets now build out the contents of the script to meet the following criteria:
• When executed the script should display the following information about your Ubuntu Server, in the order listed: ○ One-line output stating the system information about your kernel name, kernel release, and kernel version
○ Amount of free memory (RAM)
○ Storage space utilization

○ Top 5 processes utilizing memory ▪ Hint: Use the following command to view all processes running on your system: ps aux

▪ Now you will need to do a little digging on how to sort the output based on memory usage, then how to limit the output to the top 5 processes

When you have the main contents of the script figured out you will need to perform the following steps to refine the output:

• Have the script send all output created in the script to a file in the following location: /home/your_username/bin/osys2022_assign2_output.txt ○ There are multiple ways to achieve this goal so I will leave it open for you to do some research, but I will hint that some methods will require much more work than others

• Have the script create two blank lines between the output from each of the commands

• Above the output from each command you should have the script enter a line that will act as a heading explaining what is being shown in the output. Example: "Free memory available on the server"

The output of your script should look something like the image below (provided as a guide - it is okay if yours looks different, as long as it meets the criteria above). It may take some trial and error for you to get your script working as you intend.

```
Free memory available on the server
              total        used        free      shared  buff/cache   available
Mem:        4015912      165728     3586296        1208      263888     3623844
Swap:             0           0           0


Free hard drive space available on the server
Filesystem                         1K-blocks    Used Available Use% Mounted on
udev                                 1976532       0   1976532   0% /dev
tmpfs                                 401592    1208    400384   1% /run
/dev/mapper/ubuntu--vg-ubuntu--lv   4062912 2413404   1423412  63% /
tmpfs                               2007956       0   2007956   0% /dev/shm
tmpfs                                  5120       0      5120   0% /run/lock
tmpfs                               2007956       0   2007956   0% /sys/fs/cgroup
/dev/loop0                            90624   90624         0 100% /snap/core/7270
/dev/loop1                            91264   91264         0 100% /snap/core/8268
/dev/sda2                            999320  147532    782976  16% /boot
tmpfs                                401588       0    401588   0% /run/user/1000


Top processes running on the server by memory usage
USER        PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root        996  0.0  0.5 566080 24000 ?        Ssl  00:01   0:01 /usr/lib/snapd/snapd
root       1012  0.0  0.5 185948 20352 ?        Ssl  00:01   0:00 /usr/bin/python3 /usr/share/unatte
nded-upgrades/unattended-upgrade-shutdown --wait-for-signal
root        484  0.0  0.4 103052 18584 ?        S<s  00:01   0:00 /lib/systemd/systemd-journald
root       1004  0.0  0.4 169100 17244 ?        Ssl  00:01   0:00 /usr/bin/python3 /usr/bin/networkd
-dispatcher --run-startup-triggers
root        791  0.0  0.2  89860  9820 ?        Ss   00:01   0:00 /usr/bin/VGAuthService
```

Task 2 - Comments

Provide comments for each line/section of your script that meet the following criteria:

- References specific details/commands in each line
- If you provide comments in sections, be sure you reference the entirety of the section
- Speaks to the entirety of the line (example: if the line uses a pipe, you need to speak to the commands either side of the pipe)

Once your comments are complete, copy the contents of your script over into a Word document that has a title page with your full name, student #, and course code. DO NOT insert screenshots of the script as I need to copy/paste the script to test it.

Task 3 - Cron Job

With the script created you now need to configure a Cron job for the script so that it runs each morning prior to your getting to work.

Create a Cron job that meets the following criteria:

- Created under your user account
- Runs your script each morning at 6:30am, every day of the week, every day of the month ○ Take a **screenshot** showing the contents of your crontab file

Question 1: When configuring a cron job, you specify five numbers at the beginning of the job. What does each of the five numbers represent? Answer in the correct order they are in, from left to right.

Question 2: If a cron job is configured with the following numbers, provide an exact answer on when it will execute: 0 12 1 * 0

Submission

Submit the following to Brightspace:

- Document with title page that states your name, student ID and course code
- Screenshots: ○ Task 1 = 2x screenshots
  ○ Task 3 = 1x screenshots
- Answers to questions: ○ Task 3 = 2x questions
- Script: ○ Task 2 = Copy/paste contents of script into document