CompTIA Linux + Certification Week 9 Projects

Project 9- 1

In this hands-on project, you view characteristics of processes using the `ps` command.

1. Turn on your computer. After your Linux system has loaded, switch to a command-line terminal (tty2) by pressing `Ctrl+Alt+space+F2` and log in to the terminal using the user name of root and the password of `[your password]`.

2. At the command prompt, type `ps -ef | more` and press Enter to view the first processes started on the entire Linux system.

3. Fill in the following information from the data displayed on the terminal screen after typing the command:
   a. Which process has a Process ID of 1? (PID ¼ 1) ______________
   b. What character do most processes have in the terminal column (tty)? ______________
   c. What does this character in the terminal column indicate? ______________
   d. Which user started most of these processes? ______________
   e. Most processes that are displayed on the screen are started by a certain parent process indicated in the Parent Process ID column (PPID). Which process is the parent to most processes? ______________ Type q at the MORE prompt to quit.

4. At the command prompt, type `ps -el | more` and press Enter to view the process states for the first processes started on the entire Linux system.

5. Fill in the following information from the data displayed on the terminal screen after typing the command:
   a. What character exists in the State (S) column for most processes, and what does this character indicate? ______________
   b. What range of numbers is it possible to have in the Nice (NI) column? ______________
   c. Which processes have the number 4 in the Flag (F) column, and what does this number indicate? ______________ Type q at the MORE prompt to quit.
6. At the command prompt, type `ps - el | grep Z` and press Enter to display zombie processes on your Linux system. Are there any zombie processes indicated in the State (S) column?

7. Type `exit` and press Enter to log out of your shell.

**Project 9- 2**

In this hands-on project, you use `kill` signals to terminate processes on your system.

1. Switch to a command-line terminal (tty2) by pressing `Ctrl+ Alt+ space+F2` and log in to the terminal using the user name of root and the password of secret.

2. At the command prompt, type `ps - ef | grep bash` and press Enter to view the BASH shells that are running in memory on your computer. Record the PID of the BASH shell running in your terminal (tty2): __________.

3. At the command prompt, type `kill -l` and press Enter to list the available kill signals that you can send to a process.

4. At the command prompt, type `kill -2 PID` (where PID is the PID that you recorded in Step 2) and press Enter. Did your shell terminate?

5. At the command prompt, type `kill -3 PID` (where PID is the PID that you recorded in Step 2) and press Enter. Did your shell terminate?

6. At the command prompt, type `kill -15 PID` (where PID is the PID that you recorded in Step 2) and press Enter. Did your shell terminate?

7. At the command prompt, type `kill -9 PID` (where PID is the PID that you recorded in Step 2), and press Enter. Did your shell terminate? Why did this command work when the others did not?

**Project 9- 3**

In this hands-on project, you run processes in the background, kill them using the `kill` and `killall` commands, and change their priorities using the `nice` and `renice` commands.

1. Switch to a command-line terminal (tty2) by pressing `Ctrl+ Alt+ space+F2` and log in to the terminal using the user name of root and the password of secret.

2. At the command prompt, type `sleep 6000` and press Enter to start the sleep command, which waits 6,000 seconds in the foreground. Do you get your prompt back after you enter this command? Why? Send the process an INT signal by typing the `Ctrl+c` key combination.
3. At the command prompt, type `sleep 6000&` and press Enter to start the sleep command, which waits 6,000 seconds in the background. Observe the background Job ID and PID that is returned.

4. Bring the background sleep process to the foreground by typing `fg % 1` at the command prompt, and press Enter. Send the process an INT signal by typing the `Ctrl+c` key combination.

5. Place another sleep command in memory by typing `sleep 6000&` and pressing Enter. Repeat this command three more times to place a total of four sleep commands in memory.

6. At the command prompt, type `jobs` and press Enter to view the jobs running in the background. What does the + symbol indicate?

7. At the command prompt, type `kill %` and press Enter to terminate the most recent process and view the output.

8. At the command prompt, type `kill % 1` and press Enter to terminate background job # 1 and view the output.

9. At the command prompt, type `killall sleep` and press Enter to terminate the remaining sleep processes in memory. Verify that there are no more sleep processes in memory by typing the jobs command, and press Enter.

10. Place a sleep command in memory by typing `sleep 6000&` at a command prompt and pressing Enter.

11. Place a sleep command in memory with a lower priority by typing `nice -n 19 sleep 6000&` at a command prompt and pressing Enter.

12. Verify that these two processes have different nice values by typing the command `ps -el | grep sleep` at the command prompt and pressing Enter. Record the PID of the process with a nice value of 0: __________.

13. At the command prompt, type `renice + 10 PID` (where PID is the PID you recorded in the previous step) to change the priority of the process. Type the command `ps- el | grep sleep` and press Enter to verify the new priority.

14. Type exit and press Enter to log out of your shell.
**Project 9- 4**

In this hands-on project, you view and manage processes using the top command-line utility.

1. Switch to a command-line terminal (`tty2`) by pressing `Ctrl+ Alt+ Space+F2` and log in to the terminal using the user name of root and the password of secret.

2. At the command prompt, type `top` and press **Enter**.

3. From the output on the terminal top screen, record the following information:
   
   a. Number of processes: __________
   
   b. Number of sleeping processes: __________
   
   c. Amount of total memory (K): __________
   
   d. Amount of total swap memory (K): __________

4. While in the top utility, press the **h** key and observe the output. When finished, press any key to return to the previous top output.

5. By observing the output under the COMMAND column on your terminal screen, identify the PID of the top command in the output and record it: __________.

6. Type **r** in the top utility to change the priority of a running process. When asked which process to change (`renice`), type the **PID** from the previous question. When asked which value to use, type 10 to lower the priority of the top process to 10. Does this new priority take effect immediately?

7. Type **k** in the top utility to send a kill signal to a process. When asked which process, type the **PID** used in the previous question. When asked which signal to send, type 2 to send an INT signal. Did the top utility terminate?

8. At the command prompt, type `top` and press **Enter**.

9. By observing the output under the COMMAND column on your terminal screen, identify the **PID** of the top command in the output and record it: __________.

10. Type **k** in the top utility to send a kill signal to a process. When asked which process, type the PID from the previous question. When asked which signal to send, type 15 to send a **TERM** signal. Did the **TERM** signal allow top to exit cleanly?

11. At the command prompt, type **clear** and press **Enter** to clear the screen.

12. Type exit and press **Enter** to log out of your shell.
**Project 9-5**

In this hands-on project, you schedule processes by using the `at` and `cron` utilities.

1. Switch to a command-line terminal (tty2) by pressing `Ctrl+Alt+Space+F2` and log in to the terminal using the user name of root and the password of secret.

2. Schedule processes to run one minute in the future by typing the command `at now + 1 minute` at a command prompt, and press Enter.

3. When the `at>` prompt appears, type the word `date` and press Enter.

4. When the second `at>` prompt appears, type the word `who` and press Enter.

5. When the third `at>` prompt appears, press the `Ctrl+d` key combination to finish the scheduling and observe the output. When will your job run? Where will the output of the date and who commands be sent?

6. In approximately one minute, you will receive mail from the `at` daemon. Check your mail by typing `mail` at the command line and pressing Enter. Look for the e-mail with the subject “Output from your job” and record the number: __________.

7. At the `&` prompt, type the number that corresponds to the e-mail in the previous step, press Enter, and observe the output. When finished, type `q` at the `&` prompt, and press Enter to exit the mail program.

8. At the command prompt, type `cron` and press Enter to list your cron table. Do you have one?

9. At the command prompt, type `cron` and press Enter to edit a new cron table for the root user. When the vi editor appears, add the line: `30 20 * * 5 /bin/false`

10. When you finish typing, save and quit the vi editor and observe the output on the terminal screen.

11. At the command prompt, type `cron` and press Enter to list your cron table. When will the `/bin/false` command run?

12. At the command prompt, type `cat /var/spool/cron/root` and press Enter to list your cron table from the cron directory. Is it the same as the output from the previous command?

13. At the command prompt, type `cron` and press Enter to remove your cron table.

14. Type exit and press Enter to log out of your shell.
Project 9-6

In this hands-on project, you view information that is exported by the Linux kernel to the /proc directory.

1. Switch to a command-line terminal (tty2) by pressing Ctrl+Alt+Space+F2 and log in to the terminal using the user name of root and the password of secret.

2. At the command prompt, type cd /proc and press Enter to change your current directory to /proc. Next, type ls to list the directory contents and examine the output on the terminal screen. Why are the subdirectories named using numbers?

3. At the command prompt, type cat meminfo | less and press Enter to list information about total and available memory. How does the value for total memory (MemTotal) compare with the information from Step 3 in Project 9-4?

4. At the command prompt, type cat swaps and press Enter to list information about total and available swap memory. How does the value for total swap memory (Size) compare with the information from Step 3 in Project 9-4?

5. At the command prompt, type cd 1 and press Enter to enter the subdirectory that contains information about the init daemon (PID = 1).

6. At the command prompt, type ls and press Enter to list the files in the /proc/1 directory. Next, type cat status | less and press Enter. What is the state of the init daemon? Does it list the correct PID and PPID?

7. Type exit and press Enter to log out of your shell.