

Practice: Process management and monitoring commands

Here are detailed questions (with examples) on **process management and monitoring commands** in Linux:

ps – Report a snapshot of current processes

1. How would you use `ps` to display all processes running on the system, not just those owned by the current user?

Use the `ps aux` command:

```
1 ps aux
2
```

2. What is the difference between `ps -e`, `ps -f`, and `ps -ef`? Provide examples.

- `ps -e` : Displays all processes.
- `ps -f` : Provides a full-format listing with additional details.
- `ps -ef` : Combines both to show all processes in full format.

Example:

```
1 ps -e
2 ps -f
3 ps -ef
4
```

3. How can you use `ps` to find the PID of a specific process (e.g., `nginx`)?

Use the `grep` command with `ps`:

```
1 ps aux | grep nginx
2
```

4. How can you display hierarchical information about parent and child processes using `ps`?

Use the `ps --forest` option (or `--ppid` for specific parent PIDs):

```
1 ps --forest
2 ps -ef --forest
3
```

5. What is the difference between `top` and `htop`? Why might you prefer one over the other?

- `top` is a standard CLI tool but lacks interactivity.
- `htop` is more user-friendly with an interactive interface for killing processes, filtering, and sorting.

6. How can you sort the output of `top` by memory usage instead of CPU usage?

While in `top`, press `M` to sort by memory.

```
1 top
2
```

7. How can you monitor processes belonging to a specific user using `top`?

Use the `-u` option followed by the username:

```
1 top -u username
2
```

8. How can you use `htop` to search for a specific process?

Press `/` in `htop` and type the process name to search.

```
1 htop
2
```

9. Explain the significance of the `load average` displayed in `top`. What do the three numbers represent?

- Load average indicates the average number of processes waiting to run over the last 1, 5, and 15 minutes.

Example:

```
1 Load average: 1.02, 0.85, 0.77
2
```

10. How can you kill a process directly from `htop`?

Highlight the process using arrow keys and press `F9` to kill it.

Kill – Terminate processes by PID

11. What is the syntax for using `kill` to terminate a process with a known PID?

Use the `kill` command followed by the PID:

```
1 kill 1234
```

12. What does `kill -9` do? Why is it different from `kill -15` ?

- `kill -9` : Sends a SIGKILL signal to forcefully terminate the process.
- `kill -15` : Sends a SIGTERM signal, allowing the process to exit gracefully.

Example:

```
1 kill -15 1234
2 kill -9 1234
3
```

13. How can you kill all processes owned by a specific user using `kill` ?

Use `kill` with the output of the `ps` command:

```
1 kill -9 $(ps -u username -o pid=)
2
```

`pkill` – Kill processes by name

14. How is `pkill` different from `kill` ? Provide an example of each.

- `pkill` : Kills processes by name.

```
1 pkill nginx
2
```

- `kill` : Requires specifying the PID.

```
1 kill 1234
2
```

15. How can you ensure `pkill` only terminates processes owned by a specific user?

Use the `-u` option:

```
1 pkill -u username process_name
2
```

16. How can you send a specific signal (e.g., SIGHUP) using `pkill` ?

Use the `-SIGHUP` option:

```
1 pkill -SIGHUP process_name
2
```

nice / renice – Change process priority

17. What is the purpose of `nice`? How do you start a process with a lower priority?

Use `nice` followed by the priority value and command:

```
1 nice -n 10 my_program
2
```

18. How can you change the priority of a running process using `renice`?

Use the `renice` command followed by the priority and PID:

```
1 renice 5 -p 1234
2
```

19. What is the range of priority values in `nice`? What do negative and positive values signify?

- Range: `-20` (highest priority) to `19` (lowest priority).
- Negative values indicate higher priority.

20. How can you check the priority of a process?

Use the `ps -l` command:

```
1 ps -l
2
```

nohup – Run processes immune to hangups

21. What does `nohup` do, and why is it useful? Provide an example.

Runs a command immune to hangups (e.g., closing the terminal):

```
1 nohup my_program &
2
```

22. Where does `nohup` store the output of a command by default? How can you change this?

By default, output goes to `nohup.out`. You can redirect it:

```
1 nohup my_program > my_output.log &
2
```

bg / fg – Move processes to the background/foreground

23. How can you send a running process to the background?

Press `Ctrl+Z` to suspend it, then use `bg` to resume in the background:

```
1 Ctrl+Z
2 bg
3
```

24. How can you bring a background process to the foreground?

Use `fg` followed by the job number:

```
1 fg %1
2
```

25. What is the difference between backgrounding a process with `&` and using `Ctrl+Z` + `bg`?

- `&` starts a process in the background directly.
- `Ctrl+Z` + `bg` suspends and resumes a foreground process in the background.

26. How can you list all background jobs in the current shell session?

Use the `jobs` command:

```
1 jobs
2
```
