DevOps Engineering - Lab01

Lab 1 · individual or groups of 2

Themes:

1. Linux filesystem (FHS), users and permissions
2. Processes and job control
3. Pipes, I/O redirection, package management

Guidelines:

General guidelines to follow during the lab (practice ideas, not a rigid checklist):

1. Navigation and structure. On a Linux machine (VM / WSL / container), it is worth exploring `/etc`, `/var/log`, and `/proc`. A useful exercise is to identify 5 files/directories in `/etc` that are frequently used in administration, with a brief explanation of the role of each.

2. Multi-user permissions. A typical scenario: a `deploy` group, two users who are members of it, and a shared `/srv/app` directory in which both can write while others can only read (`chmod`, `chown`, `chmod g+s`). Checking with `ls -la` and performing a write test from both accounts helps with understanding.

3. Pipes on logs. From `/var/log/auth.log` (or a sample log), a useful pipeline is `grep | awk | sort | uniq -c | sort -nr` to obtain the top 5 IP addresses with the most failed authentication attempts.

4. Process management. Start a process in the background, monitor it with `ps/top`, then stop it with `SIGTERM` and (on a new process) with `SIGKILL`, followed by a brief note on the difference (e.g. in `signals.txt`).

5. File cleanup. A useful one-liner deletes files older than 7 days from a test `tmp/` directory without affecting subdirectories marked as important (with the exceptions documented).

References:

1. Filesystem Hierarchy Standard: https://refspecs.linuxfoundation.org/FHS_3.0/fhs/index.html
2. GNU Coreutils (chmod, chown, ls): https://www.gnu.org/software/coreutils/manual/
3. signal(7) - Linux man-pages: https://man7.org/linux/man-pages/man7/signal.7.html
4. The Linux Command Line (optional): https://linuxcommand.org/tlcl.php

What is useful to keep. Commands + relevant output + brief explanations; when working in a team, a `roles.txt` file helps clarify contributions.

AI note. If you use AI during the lab, it is useful to briefly note how you used it (what you manually verified, advantages/disadvantages), so that it remains clear what you understood versus what was generated.