Operating Systems Lab 3

Tarek Abdelbari SIBACHIR

January 2025

1 Introduction

The purpose of this lab is to explore and understand the Linux directory structure, as well as to practice using command-line tools and aliases effectively.

2 Task 1: Understanding Linux Directories

The following is a brief overview of important Linux directories:

1. `/bin`: Contains essential binary executables necessary for all users, including basic commands like `ls`, `cp`, and `mv`.
2. `/sbin`: Contains system binaries primarily for the root user, including commands for system administration such as `shutdown` and `mount`.
3. `/boot`: Contains files required for the boot process, including the Linux kernel and initial RAM disk images.
4. `/dev`: Contains device files that represent hardware and virtual devices, allowing software to interact with hardware components.
5. `/etc`: Contains configuration files and settings for the system and installed applications, including user account information and system-wide settings.
6. `/lib`: Contains shared library files essential for the binaries in `/bin` and `/sbin` to function properly.
7. `/media`: Contains mount points for removable media such as USB drives, CDs, and DVDs, typically mounted automatically.
8. `/mnt`: A temporary mount point for mounting filesystems, often used by system administrators for manual mounting of devices.
9. `/opt`: Contains optional software packages that are not part of the default installation, often used for third-party applications.

10. `/proc`: A virtual filesystem that provides an interface to kernel data structures, allowing users to access information about system processes and hardware.

11. `/root`: The home directory of the root user, which is the administrative account in Linux.

12. `/run`: A temporary filesystem that stores runtime information, such as system information and process IDs, which is cleared on reboot.

13. `/srv`: Contains data for services provided by the system, such as web or FTP servers, typically organized by service type.

14. `/sys`: A virtual filesystem that exposes kernel information and allows interaction with kernel subsystems, devices, and other kernel-related data.

15. `/tmp`: A temporary directory used for storing transient files created by applications; files in this directory are typically deleted on reboot.

16. `/usr`: Contains user-related programs and data, including applications, libraries, and documentation; it is often further divided into subdirectories like `/usr/bin` and `/usr/lib`.

17. `/var`: Contains variable data files, such as logs, databases, and spool files, which are expected to change in size and content over time.

18. `/home`: The directory where user-specific files and personal data are stored, including documents, photos, and downloads for each user.

2.1 Displaying the Output of the Alias Command

2.2 Creating a New Alias

I have created the following alias:

```
alias psa="ps -aux"
```

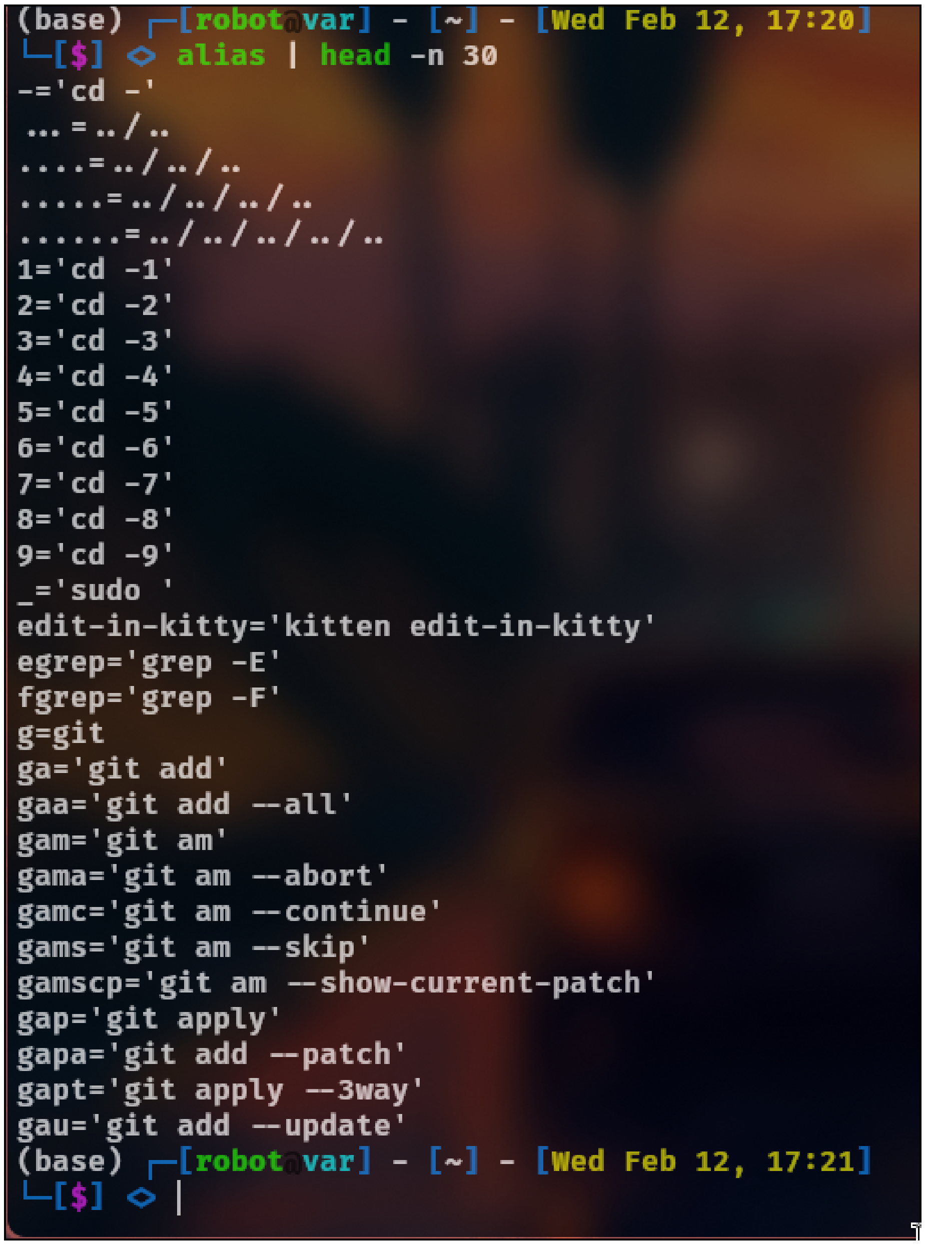
2.3 Adding the Alias to the .zshrc File

To make the alias permanent, I added it to the `.zshrc` file by executing:

```
echo 'alias psa="ps -aux"' >> ~/.zshrc
```

Then, I ran the following command to load the new configuration:

```
source ~/.zshrc
```



```
(base) ┌─[robot@var] - [~] - [Wed Feb 12, 17:20]
└─[$] <> alias | head -n 30
-='cd -'
...=../..
....=../../..
.....=../../../..
......=../../../../..
1='cd -1'
2='cd -2'
3='cd -3'
4='cd -4'
5='cd -5'
6='cd -6'
7='cd -7'
8='cd -8'
9='cd -9'
_='sudo '
edit-in-kitty='kitten edit-in-kitty'
egrep='grep -E'
fgrep='grep -F'
g=git
ga='git add'
gaa='git add --all'
gam='git am'
gama='git am --abort'
gamc='git am --continue'
gams='git am --skip'
gamscp='git am --show-current-patch'
gap='git apply'
gapa='git add --patch'
gapt='git apply --3way'
gau='git add --update'
(base) ┌─[robot@var] - [~] - [Wed Feb 12, 17:21]
└─[$] <> |
```

Figure 1: Output of the alias command

2.4 Executing New Commands

To run commands in the terminal, you can use the following methods:

- **Sequential Commands**: Use a semicolon to separate commands. For example:

  ```
  git clone https://github.com/TarekAeb/Ensia; cd Ensia
  ```

- Conditional Commands: Use to run the second command only if the first command succeeds. For example:

  ```
  yt-dlp https://www.youtube.com/watch?v=tPZauAYgVRQ && mpv " Elon Musk attempts host
  ```

- Background Execution: Use at the end of the command to run it in the background. For example:

  ```
  sudo docker compose up -d
  ```