2. Student Handout

Linux Basics and Commands: Student Handout

---

1. Overview of Linux

1.1 History of Linux

- **Linux Creation**: Developed by Linus Torvalds in 1991 as a free and open-source OS.
- **UNIX Inspiration**: Modeled after UNIX, used in academic and industrial settings.
- **Growth**: Now widely used in servers, cloud computing, and embedded systems.

1.2 Linux Distributions

- **Ubuntu**: User-friendly with large community support.
- **Fedora**: Features the latest software.
- **CentOS**: Focused on stability for enterprise use.
- **Debian**: Known for stability and extensive software repository.

1.3 Importance of Linux in the Industry

- **Servers**: Powers over 90% of the world's servers.
- **Cloud Computing**: Integral to AWS, Azure, and Google Cloud.
- **Embedded Systems**: Used in devices like routers and smart TVs.
- **Supercomputers**: Runs almost all top 500 supercomputers.

---

2. Understanding the Linux File System Hierarchy

- **Root Directory (`/`)**: The starting point of the Linux file system.

- **Key Directories**:
  - **/bin**: Essential user binaries.
  - **/boot**: Boot files, including the Linux kernel.
  - **/dev**: Device files for hardware components.
  - **/etc**: System and software configuration files.
  - **/home**: Personal directories for users.
  - **/lib**: Shared libraries for `/bin` and `/sbin`.
  - **/tmp**: Temporary files.
  - **/usr**: User programs and utilities.
  - **/var**: Variable data like logs and databases.

---

3. Basic Linux Commands

3.1 Navigating the File System

- **pwd**: Displays the current directory.
  - Example: `$ pwd` → `/home/john`
- **ls**: Lists files and directories.
  - Example: `$ ls` → `Documents Downloads Pictures`
- **cd**: Changes the current directory.
  - Example: `$ cd Documents`
  - Example: `$ cd ..` (moves up one level)

3.2 Managing Files and Directories

- **touch**: Creates an empty file.
  - Example: `$ touch file.txt`
- **mkdir**: Creates a new directory.
  - Example: `$ mkdir new_folder`
- **cp**: Copies files or directories.
  - Example: `$ cp file.txt new_folder/`
- **mv**: Moves or renames files or directories.
  - Example: `$ mv file.txt new_folder/`
  - Example: `$ mv file.txt new_file.txt`
- **rm**: Removes files or directories.

- Example: `$ rm file.txt`
- Example: `$ rm -r new_folder` (removes directory and contents)

3.3 Viewing and Editing Files

- **cat**: Displays file contents.
  - Example: `$ cat file.txt`
- **nano**: Opens a file in the nano text editor.
  - Example: `$ nano file.txt`

---

4. Potential Gaps or Unclear Points

- **Linux vs. Windows**: Linux primarily uses the command line, unlike Windows' graphical interface.
- **Permissions**: Linux has a strict permission system for file access.
- **Case Sensitivity**: Linux distinguishes between `File.txt` and `file.txt`.

---

5. Diagrams to Help Understand Concepts

5.1 Linux File System Hierarchy

```
/
├── bin
├── boot
├── dev
├── etc
├── home
│   ├── john
│   └── jane
├── lib
├── opt
├── tmp
├── usr
└── var
```

5.2 Basic Command Flow

```
Start
  |
  v
[pwd] --> Shows current directory
  |
  v
[ls] --> Lists files in the current directory
  |
  v
[cd] --> Changes directory
  |
  v
[mkdir] --> Creates a new directory
  |
  v
[touch] --> Creates a new file
  |
  v
[cp] --> Copies a file
  |
  v
[mv] --> Moves or renames a file
  |
  v
[rm] --> Deletes a file
```

Conclusion

This handout provides a foundational understanding of Linux, its file system, and basic commands. Practice these commands in a Linux environment to become more comfortable with the system. Happy learning!