

Note: Setup Home Lab

Table of Contents

1	System Base	1
2	Common Linux Directory Structure	1
3	General Preparation of Linux system	2
3.1	Update and Upgrade System	2
3.2	Install essential tools:.....	2
4	Install Docker.....	3
5	More is coming soon.....	4

1 System Base

- Raspberry Pi 5
- Ubuntu Desktop

2 Common Linux Directory Structure

- / (Root): The base of the file system. All files and directories start from here.
- /bin: Contains essential user binaries (executables), such as commonly used commands and programs. Examples include ls, cp, and mv.
- /boot: Holds files needed to start up the system, including the Linux kernel, an initial RAM disk image (for drivers needed at startup), and the bootloader.
- /dev: Contains device files that represent and provide access to hardware components (like disks, terminals, or any physical or virtual device).
- /etc: This directory contains configuration files required by all programs. This is where the host-specific system-wide configuration files are stored.
- /home: Contains the personal directories of all users who have accounts on the system, except the root user.
- /lib: Includes essential shared libraries and kernel modules needed by the system and the programs running on it. Libraries needed to boot the system and run the commands in /bin and /sbin are here.
- /media: A mount point for removable media such as USB drives, CD-ROMs, etc.
- /mnt: Temporary mount points for mounting filesystems manually.
- /opt: Optional or third-party software. It is often used to install software not included with the distribution.
- /proc: A virtual filesystem that provides a mechanism for kernel to send information to processes. It contains information about system resources and the status of the operating system.

- `/root`: The home directory of the root user (the system administrator). It is not located under `/home` to ensure that the root user can still log in even if `/home` is not available.
- `/sbin`: Contains system binaries, i.e., executables that perform tasks required for system maintenance at a higher level than those in `/bin`.
- `/srv`: Contains data served by the system, such as web files and FTP data.
- `/tmp`: A directory for storing temporary files, which are cleared at reboot.
- `/usr`: This is one of the largest directories and contains multi-user applications, libraries, documentation, etc. It holds the majority of user utilities and applications.
- `/var`: Variable data files. This includes spool directories and files, administrative and logging data, and transient and temporary files.

3 General Preparation of Linux system

3.1 Update and Upgrade System

❖ Run

➤ `sudo apt update && sudo apt upgrade`

- `sudo`: This is a command that allows you to run programs with the security privileges of another user, by default the superuser or root. It's used here to ensure that the command has the necessary permissions to modify system files.
- `apt`: This is the package handling utility in Ubuntu and other Debian-based Linux systems. It stands for "Advanced Package Tool."
- `update`: This command updates the list of available packages and their versions, but it does not install or upgrade any packages. It essentially syncs the database of available software packages with the repositories specified in `/etc/apt/sources.list` and its additional configuration files in `/etc/apt/sources.list.d/`. This command is crucial for informing the system about the newest available versions of packages and any dependencies they might have.
- `&&`: This is a logical operator used in shell commands to combine two commands. The second command (`sudo apt upgrade` in this case) will only execute if the first command (`sudo apt update`) completes successfully. This ensures that the upgrade process only attempts to proceed if the update was successful and the package lists are fresh.
- `upgrade`: This command upgrades all the installed packages to the latest versions (according to the updated lists fetched with `apt update`). It will automatically handle the installation of updated versions of currently installed packages and any required dependencies. It does not, however, automatically install new packages or remove existing packages.
- Second `sudo` and `apt` are used in the same way as explained above.
- Overall, running `sudo apt update && sudo apt upgrade` is a good practice for maintaining the security and performance of your system, ensuring all software is up-to-date with the latest features and security patches.

3.2 Install essential tools:

❖ Run

➤ `sudo apt install vim git curl`

- `sudo`: This is a command that allows you to run programs with the security privileges of another user, by default the superuser or root. It is necessary here because installing software typically requires administrative privileges.

- apt: Short for "Advanced Package Tool", this is the package management utility used in Debian-based systems for handling the installation, updating, and removal of software.
- install: This command tells APT to install the specified packages.
- vim, git, curl: These are the names of the packages to be installed.
- Below's a brief overview of each package:
 - vim: This is a highly configurable text editor built to enable efficient text editing. It is an improved version of the vi editor distributed with most UNIX systems. Vim is often used by programmers and system administrators and is known for its power and flexibility.
 - git: Git is a distributed version control system used to track changes in source code during software development. It is designed for coordinating work among programmers, but it can be used to track changes in any set of files. It is a crucial tool for developers, allowing for robust version control of projects.
 - curl: Curl is a command-line tool used for transferring data with URLs. It supports numerous protocols, including HTTP, HTTPS, FTP, and more. It's widely used for testing, downloading files, making API requests, and automating interactions with websites.
- By running this command, you are essentially setting up your system with a powerful text editor, a version control system, and a tool for data transfer over networks, which can be particularly useful for software development and system administration tasks.

4 Install Docker

Before you install Docker Engine for the first time on a new host machine, you need to set up a Docker's official apt repository.

Docker's official APT repository is a software repository managed by Docker, Inc. that provides the packages necessary to install Docker on Debian-based operating systems, including Ubuntu. This repository is specifically designed to allow users to install and maintain Docker using the native package management tools like 'apt' (Advanced Package Tool).

❖ Why Use Docker's Official APT Repository?

- There are several advantages to using Docker's official APT repository over the default repositories available in Ubuntu or other Debian-based distributions:
 - Latest Versions: Docker's repository often includes more recent versions of Docker than what might be available in the default Ubuntu repositories. This is particularly important for accessing new features and security updates.
 - Stability and Compatibility: The versions provided through Docker's repository are tested and vetted by Docker, ensuring that they work as intended on supported systems.
 - Security: Docker signs its packages and the repository with a GPG key, allowing users to verify the integrity and authenticity of the packages they download and install.

❖ How It Works:

- When you set up Docker's official APT repository, you perform actions to ensure your system can securely fetch and install Docker from this source:
 - Install HTTPS support for APT: This is necessary because the repository is accessed over HTTPS to ensure that all communications and downloads are secure.
 - Add Docker's GPG Key: This public key is used to verify that the packages downloaded from the repository are signed by Docker and have not been tampered with.
 - Configure the repository source: You add entries to your system's package source list that point to Docker's official repository. This tells apt where to find Docker packages.

- Update and install: Once the repository is configured, you update your package list with apt update, and then you can install Docker using apt install, knowing that you are getting the software directly from Docker.
- ❖ This setup ensures that you're always able to install the latest official Docker releases using your system's standard package management tools, keeping Docker up to date easily and securely.

5 More is coming soon.....