

Using Linux for Self-Hosting

[Linux Training Academy](#)

Lesson Overview

- What Linux Is
- Why Linux
- Why Ubuntu LTS

Linux is a Kernel

- **Linux** is technically a **kernel**
- A **kernel** is the **core** of an OS that interacts with hardware
- It sits between hardware and applications
- Acts as an interface for applications to use hardware (e.g., processors, memory)
- Intermediary between software and hardware

Linux Is Also an Operating System

- **Linux** often refers to a **Linux distribution**
- A Linux distribution = Linux kernel + software
- Operating System (OS): manages hardware and application execution
- Enables storage, display, and networking capabilities
- Examples of Operating Systems: Windows & macOS

Linux Distributions (Distros)

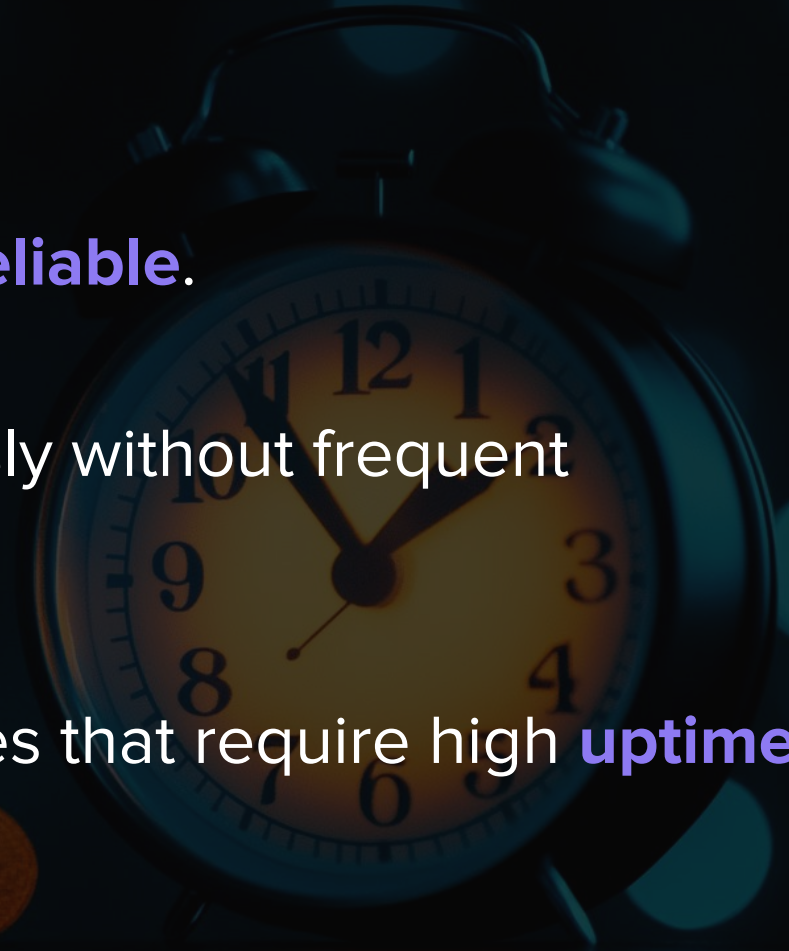
- Linux distribution = Linux kernel + software
- Often called "**distros**" or "**flavors**"
- Different distros come with different default software
- Example: Some distros use Nano, others use Vim as default editors
- **Customizable**: Users can install preferred software
- Linux distros are **curated collections** of compatible software

Open Source and Free

- Linux is **open-source** and **free** to use.
- No licensing fees, making it **cost-effective**.
- Ideal for self-hosted environments that aim to reduce costs.

Stability and Reliability

- Linux is highly **stable** and **reliable**.
- Designed to run continuously without frequent crashes.
- Essential for hosting services that require high **uptime**.



Security

- Open-source nature allows for **fast vulnerability fixes**.
- **Built-in security** features: firewall, strict permissions, and process isolation.
- Tools like Docker improve security through application containment.
- **Security is an ongoing process**—requires updates and vigilance.

Extensive Software Availability and Support

- Thousands of open-source applications designed for **Linux**.
- Many self-hosted **alternatives** to cloud-based services.
- **Strong communities** for troubleshooting and support.

Self-Hosting with Docker

- Docker packages applications into portable **containers**.
- Eliminates compatibility issues (version mismatches, missing libraries).
- Ensures **consistent performance** across environments.
- Easier deployment and maintenance of self-hosted services.

Choosing a Linux Distribution for Self-Hosting

- Linux is **open-source**, allowing anyone to create a distribution.
- Over **1,000** Linux distributions exist, but fewer than **300** are actively maintained.
- Only a small number of distros are widely used and recommended.
- **Docker** compatibility is a key factor in choosing a distribution for self-hosting.

Distributions Officially Supported by Docker

- Docker officially supports:
 - CentOS
 - Debian
 - Fedora
 - Raspberry Pi OS
 - Red Hat Enterprise Linux (RHEL)
 - SUSE Linux Enterprise Server (SLES)
 - Ubuntu
- Docker may work on other distros, but will have limited support if issues arise.

Why **Ubuntu** Is a Good Choice for **Self-Hosting**

- Ubuntu is recommended for self-hosting due to:
 - Stability
 - Ease of use
 - Extensive community support
- **Ubuntu LTS** (Long Term Support) releases provide extended updates and security patches.

Long Term Support (LTS) Releases of Ubuntu

- LTS versions receive **5 years** of security updates.
- Example: Ubuntu 24.04 LTS (April 2024) will be supported until April 2029.
- Ubuntu releases versions every 6 months, but non-LTS versions are only supported for 9 months.
- LTS versions are best for self-hosting, ensuring **long-term stability**.

Documentation and Community Support

- **Ubuntu** has one of the most active Linux **communities**.
- Extensive documentation, guides, and tutorials available.
- Many Docker-related resources are written for Ubuntu.
- **Widely used** in business environments, ensuring long-term viability.

Windows Subsystem for Linux (WSL) Support

- Ubuntu is the default Linux distribution for **WSL**.
- Easily install and run Ubuntu on Windows.
- Best of both worlds: Use Windows while benefiting from Ubuntu.

Bare Metal vs. Virtual Machine (VM) Setups

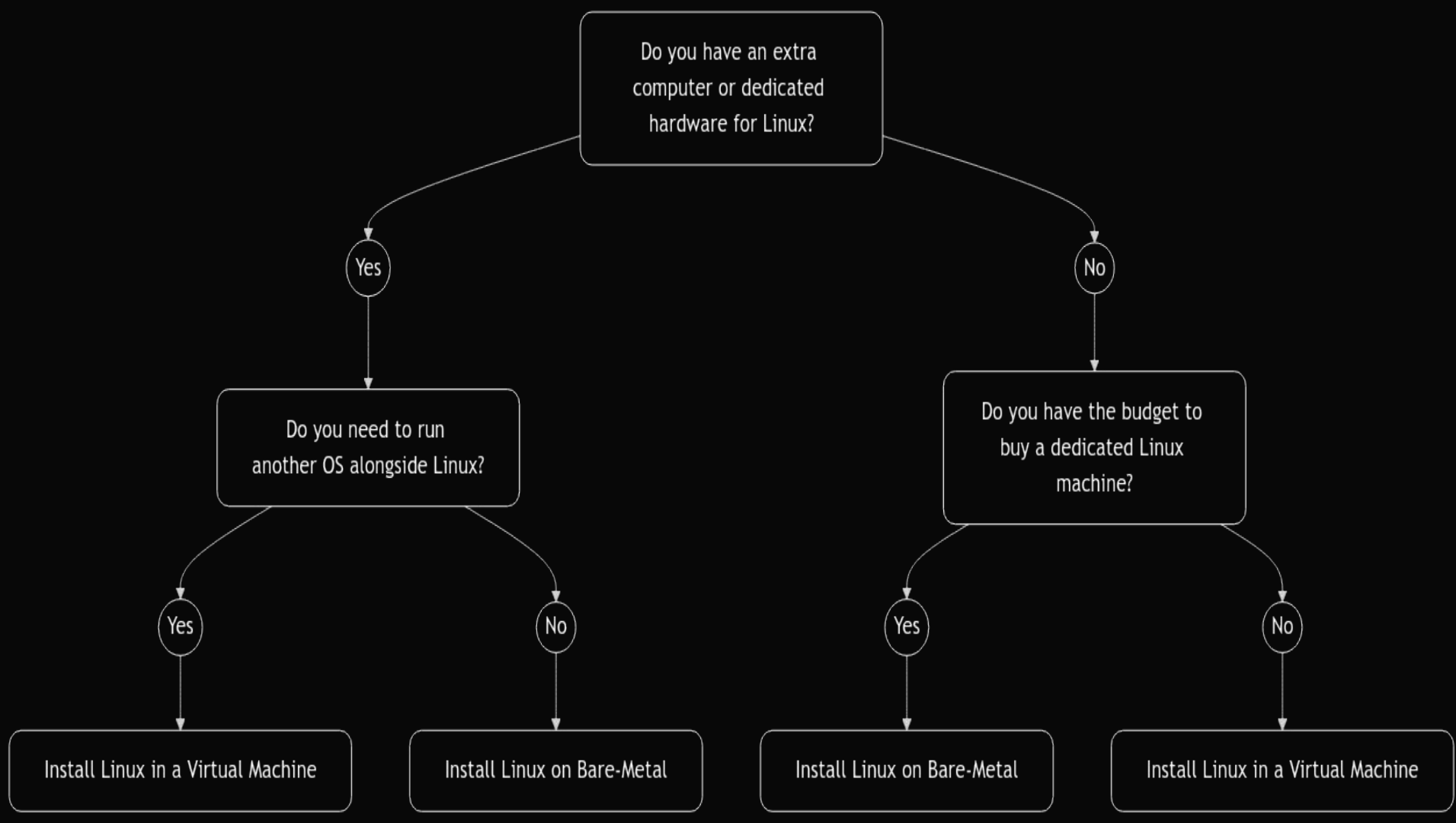
- Two main options for deploying Linux for self-hosting:
 - **Bare Metal**: Direct installation on physical hardware.
 - Virtual Machine (**VM**): Runs within a software-based environment.
- Each option has advantages based on performance, convenience, and hardware availability.

Linux on Bare Metal

- Installed **directly** onto physical hardware.
- Best **performance** – full access to CPU, memory, and storage.
- No virtualization overhead = faster and more **efficient**.
- More **stable** and reliable with fewer moving parts.

Linux in a Virtual Machine (VM)

- Linux runs as a **guest** OS inside a host OS (e.g., Windows, macOS).
- **Convenient** for testing and learning without replacing the host OS.
- **Shares** hardware resources with the host.
- Adds a virtualization layer, slightly impacting performance.
- Still a great choice for self-hosting.



Lesson Recap

- Linux is a kernel, but the term often refers to a Linux distribution (OS).
- Linux is ideal for self-hosting because it is free, open-source, and stable.
- Many self-hosted applications use Docker for isolation and easy deployment.
- Ubuntu LTS is recommended for its Docker support, strong community, and long-term updates.
- Linux can be installed on bare metal or in a virtual machine.

Next Step

- Go to the lesson that matches your Linux setup choice:
 - **Installing Ubuntu on Windows** → If using a virtual machine on Windows.
 - **Installing Ubuntu on macOS** → If using a virtual machine on macOS.
 - **Installing Ubuntu on Bare Metal** → If installing directly on dedicated hardware.