# 010123131

# Software Development Practice I

## Handout #1

<rawat.s@eng.kmutnb.ac.th>

Last Update: 2024-06-29

# Expected Learning Outcomes (1)

Students are expected to be able to:

- Explain the benefits of using **Linux**.
- Describe different options for using **Linux environments**.
- Install **Linux distributions** in **virtual machines** (**VMs**).
- Set up **WSL2** for a **Microsoft Windows 10 / 11 systems**.
- Learn & use some **basic Linux commands**.

# Expected Learning Outcomes (2)

Students are expected to be able to:

- Write **Linux Bash shell scripts for system administration**.
- Write and compile **C/C++ source code files** or build embedded software projects using an open source cross-platform toolchain.
- Install Linux OS such as Raspberry Pi OS or Ubuntu on a **single-board computer (SBC)** such as the **Raspberry Pi 3 or 4 or 5.**
- Install programs / setup some services on a Linux-based system.
- [***Mini-Project***] Develop a showcase for **Linux-based home automation / smart home applications**.

# Linux

- Linux is a family of **open-source Unix-like operating systems** based on the **Linux Kernel**.
  - Ref.: https://en.wikipedia.org/wiki/Linux

- Linux can be installed on different types of computers and supports various processor architectures (e.g. x86, ARM Cortex-A, RISC-V, …)

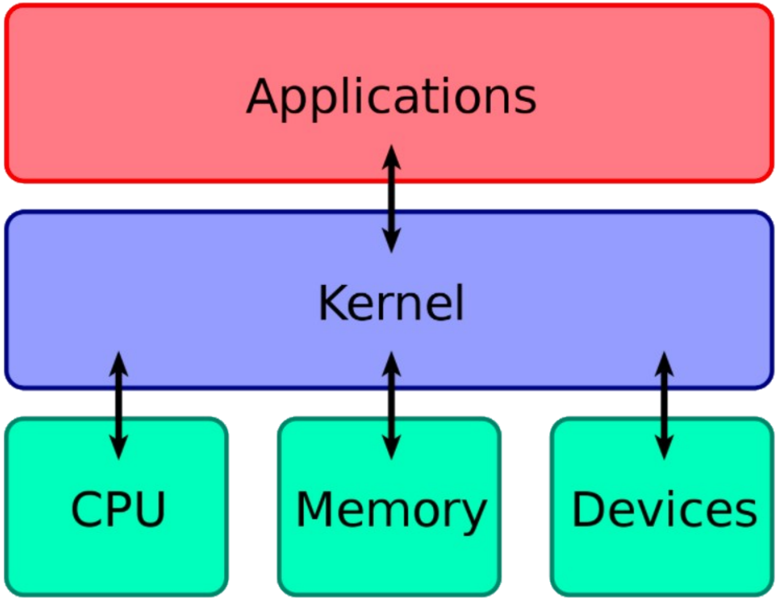
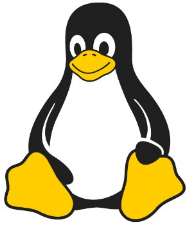


Image source: https://en.wikipedia.org/wiki/Kernel_(operating_system)

# Linux Distributions



- Linux is typically packaged in a **Linux Distribution** (or **Linux Distro** for short) which includes the **Linux kernel** and supporting system software and libraries, bundled as a single **image file** (.img or .iso).
- Popular **Linux distributions** are **Debian**, **Ubuntu**, **OpenSUSE**, **Fedora**, **Arch Linux**, **CentOS**, ....
- Commercial distributions include **Red Hat Enterprise Linux** and **SUSE Linux Enterprise (SLE)**.

# Is it worth learning Linux ?

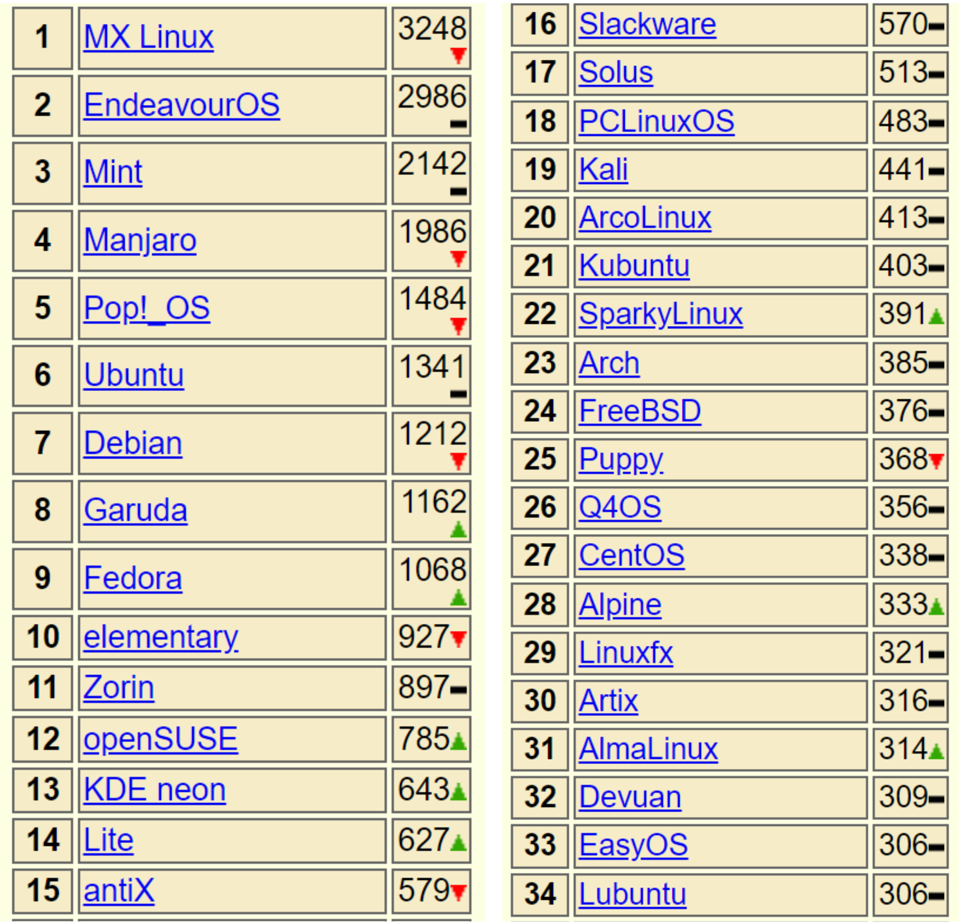
## *Here are some statistics about Linux in 2021.*

- Only two out of the 25 **most popular websites worldwide** (or 8%) don’t use Linux.
- In 2020, 54.2% of the **most powerful supercomputers** operated on Linux.
- According to 83.1% of **professional developers**, Linux is **the most loved platform**.
- About 90% of the **public cloud workload** is run by Linux.
- On the **Amazon EC2 cloud computing platform**, Linux controls 92% of the market.

**Source: "Linux Statistics That Will Completely Amaze You" (June 2021)**
https://writersblocklive.com/blog/linux-statistics/

- According to "**DistroWatch Page Hit Ranking**", the 3 most popular Linux distributions are **MX Linux**, **Manjaro** and **Linux Mint**.
- **Ubuntu** is on the 6th place, followed by **Debian**.

Note: The third column shows the integer number of times a distribution page on **DistroWatch.com** was accessed each day, for the specified period (e.g. the last 12 months).

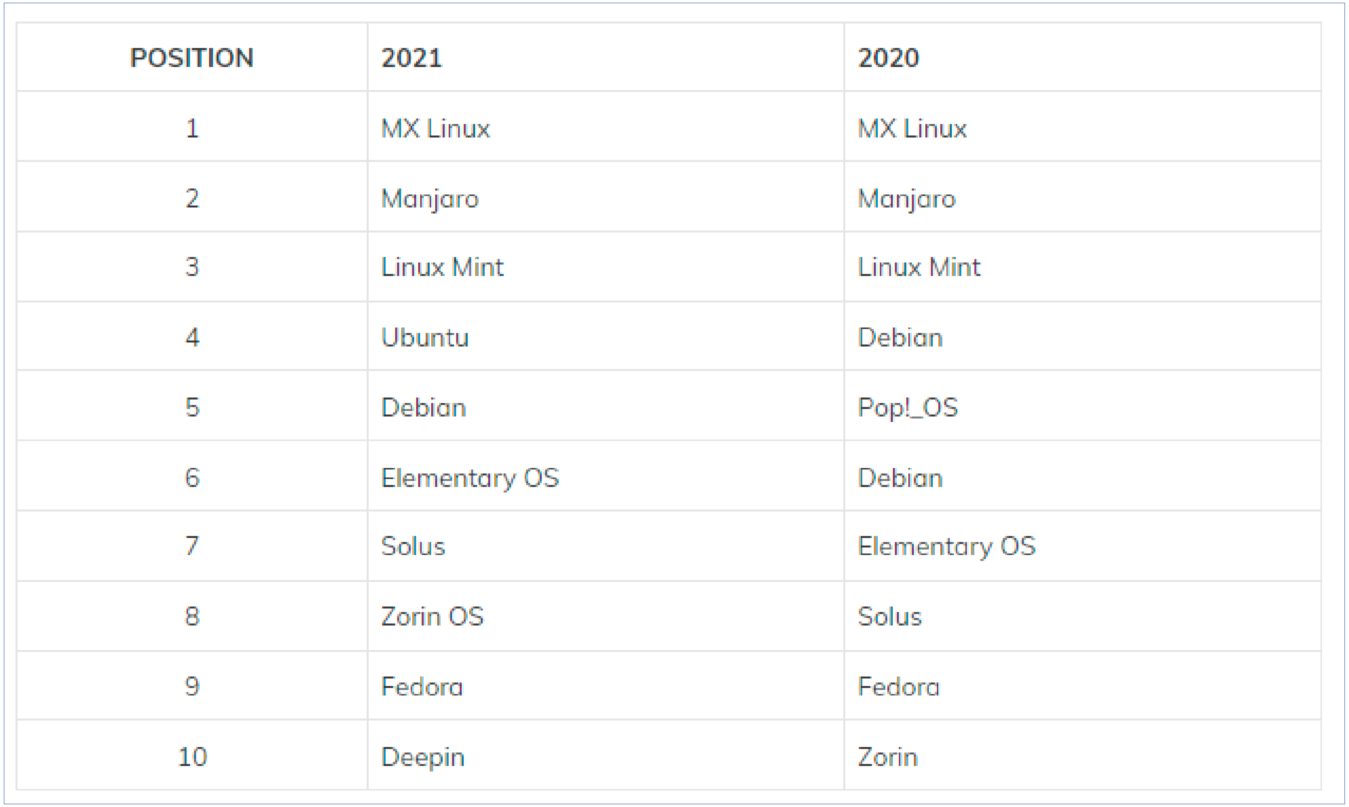


| | | |
|---|---|---|
| 1 | MX Linux | 3248▼ |
| 2 | EndeavourOS | 2986▬ |
| 3 | Mint | 2142▬ |
| 4 | Manjaro | 1986▼ |
| 5 | Pop!_OS | 1484▼ |
| 6 | Ubuntu | 1341▬ |
| 7 | Debian | 1212▼ |
| 8 | Garuda | 1162▲ |
| 9 | Fedora | 1068▲ |
| 10 | elementary | 927▼ |
| 11 | Zorin | 897▬ |
| 12 | openSUSE | 785▲ |
| 13 | KDE neon | 643▲ |
| 14 | Lite | 627▲ |
| 15 | antiX | 579▼ |
| 16 | Slackware | 570▬ |
| 17 | Solus | 513▬ |
| 18 | PCLinuxOS | 483▬ |
| 19 | Kali | 441▬ |
| 20 | ArcoLinux | 413▬ |
| 21 | Kubuntu | 403▬ |
| 22 | SparkyLinux | 391▲ |
| 23 | Arch | 385▬ |
| 24 | FreeBSD | 376▬ |
| 25 | Puppy | 368▼ |
| 26 | Q4OS | 356▬ |
| 27 | CentOS | 338▬ |
| 28 | Alpine | 333▲ |
| 29 | Linuxfx | 321▬ |
| 30 | Artix | 316▬ |
| 31 | AlmaLinux | 314▲ |
| 32 | Devuan | 309▬ |
| 33 | EasyOS | 306▬ |
| 34 | Lubuntu | 306▬ |

**Last Access on: 2022-06-15**

https://distrowatch.com/dwres.php?resource=popularity

# “10 Top Most Popular Linux Distributions of 2021” (May 31, 2021)



| POSITION | 2021 | 2020 |
|---|---|---|
| 1 | MX Linux | MX Linux |
| 2 | Manjaro | Manjaro |
| 3 | Linux Mint | Linux Mint |
| 4 | Ubuntu | Debian |
| 5 | Debian | Pop!_OS |
| 6 | Elementary OS | Debian |
| 7 | Solus | Elementary OS |
| 8 | Zorin OS | Solus |
| 9 | Fedora | Fedora |
| 10 | Deepin | Zorin |

Source: https://www.tecmint.com/top-most-popular-linux-distributions/

# Debian Code names

- Code names for different versions of **Debian**:
  - Debian 9 or "Stretch" (released in 2017)
  - Debian 10 or "Buster" (released in 2019)
  - Debian 11 or "Bullseye" (released in 2021)
  - Debian 12 or “Bookworm” (released in 2023)

# Ubuntu Code Names

- Code names for **Ubuntu** releases:
  - Ubuntu 18.04 LTS or "Bionic Beaver" (released in 2018)
  - Ubuntu 20.04 LTS or "Focal Fossa" (released in 2020)
  - Ubuntu 22.04 LTS or "Jammy Jellyfish" (released in 2022)
  - Ubuntu 24.04 LTS or “Noble Numbat” (released in 2024)

Notes:

- The first release was Ubuntu 4.10, released in October 2004.
- Ubuntu is released every 6 months, with long-term support (LTS) releases every 2 years.
- Ubuntu LTS releases are the 'enterprise grade' releases of Ubuntu.

https://wiki.ubuntu.com/Releases

# Approaches to learn Linux

- Installing Linux on a physical machine (SSD, NVMe)
- Using a Linux live CD or USB drive or MicroSD
- Running Linux on a virtual machine (VM)
- Using Cloud-based Linux virtual machines
- Using a low-cost single-board computers such as Raspberry Pi boards

# Approaches to learn Linux

- Linux can be installed directly on a physical computer or server.
- This allows for a more immersive learning experience as you get to work with the hardware directly.
- However, this approach may be limited by the availability of physical hardware and the risk of damaging it if not done correctly.

# Approaches to learn Linux

- Running Linux on VM software such as VirtualBox is another approach.
- This allows for a more flexible and safe learning experience, as the user can easily create, modify, backup and delete virtual machines without affecting the host OS.
- With proper hardware resources such as memory and storage, multiple VMs can run on the same physical hardware, allowing for more efficient use of resources.

# Approaches to learn Linux

- Another approach involves booting into a Linux operating system from a CD or USB drive.
- Cloud service providers such as Amazon Web Services or Microsoft Azure or Digital Ocean allow users to create and manage virtual machines running Linux.
- Using a low-cost single-board computer, such as the Raspberry Pi, is another great approach to learning Linux.

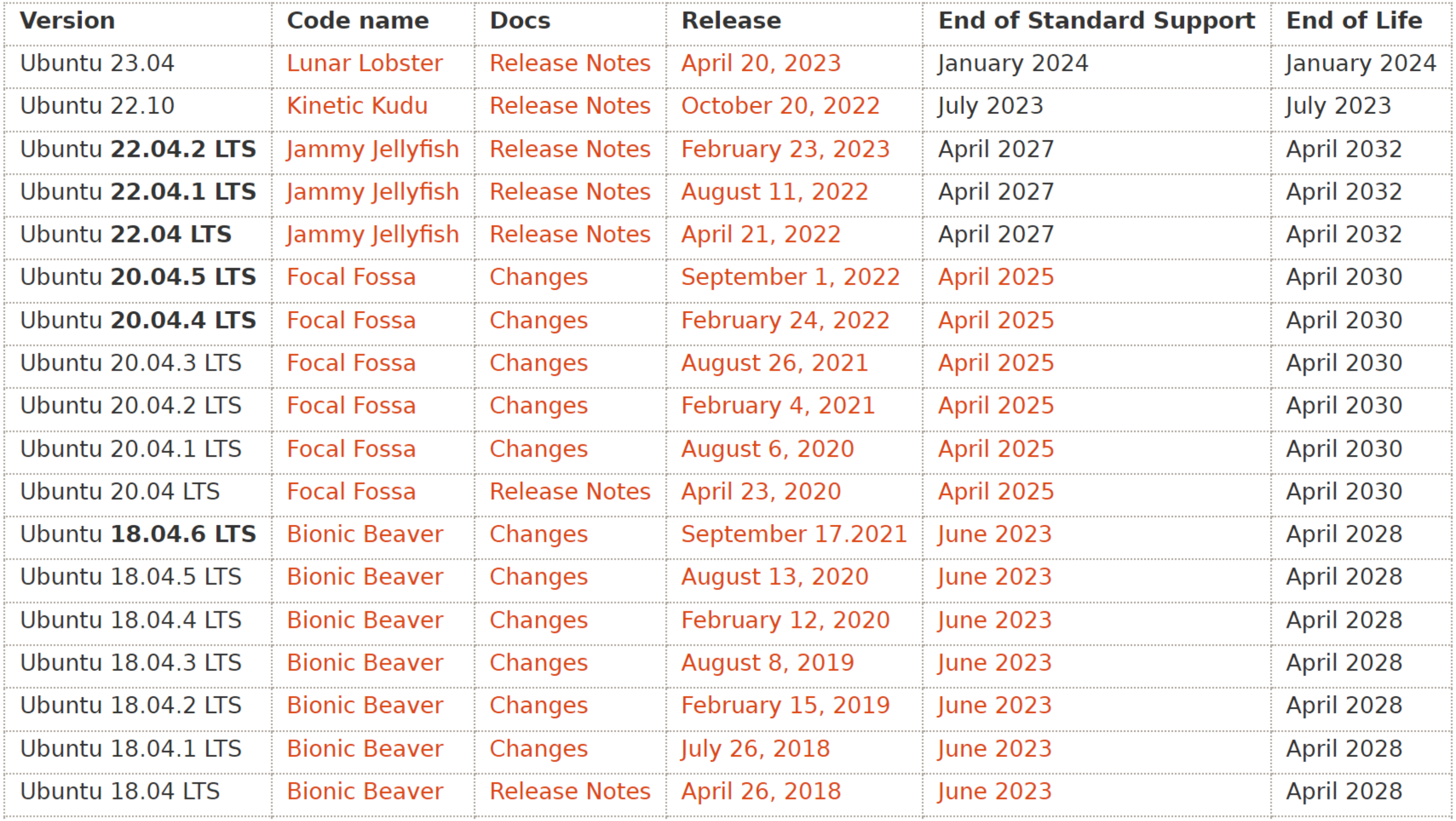
# Ubuntu Server vs. Ubuntu Desktop

- The **Ubuntu Server Edition** and the **Ubuntu Desktop Edition** use the same “apt” repositories, making it just as easy to install a server application on the Desktop Edition as on the Server Edition.
- ***One major difference is that the graphical environment used for the Desktop Edition is not installed for the Server edition***. This includes the graphics server itself, the graphical utilities and applications, and the various user-supporting services needed by desktop users.

**Basic installation for Ubuntu Server Edition**
https://ubuntu.com/server/docs/installation

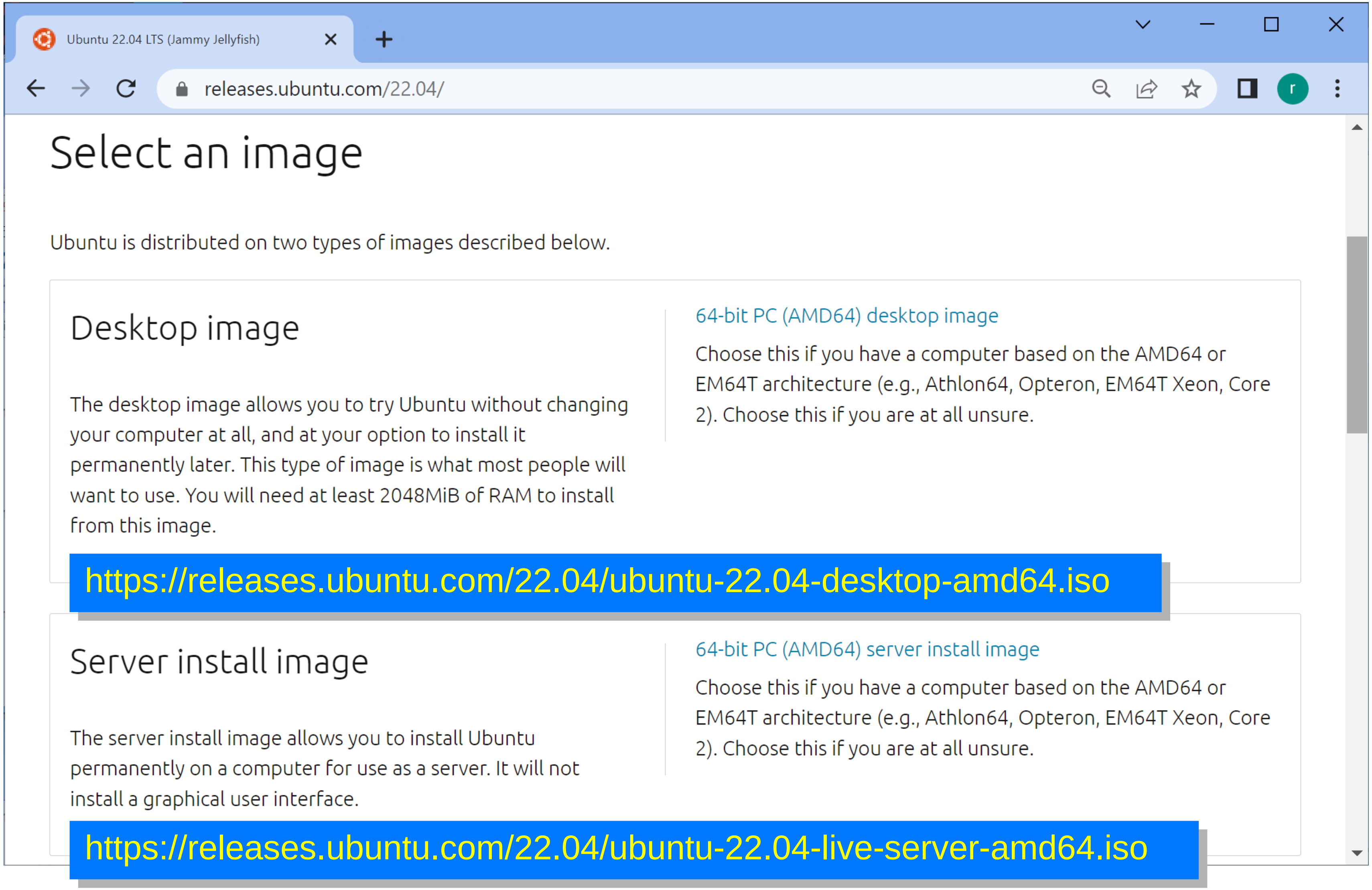
**Ubuntu 22.04 LTS (Jammy Jellyfish)**
https://releases.ubuntu.com/22.04/

# List of Ubuntu releases



| Version | Code name | Docs | Release | End of Standard Support | End of Life |
|---|---|---|---|---|---|
| Ubuntu 23.04 | Lunar Lobster | Release Notes | April 20, 2023 | January 2024 | January 2024 |
| Ubuntu 22.10 | Kinetic Kudu | Release Notes | October 20, 2022 | July 2023 | July 2023 |
| Ubuntu **22.04.2 LTS** | Jammy Jellyfish | Release Notes | February 23, 2023 | April 2027 | April 2032 |
| Ubuntu **22.04.1 LTS** | Jammy Jellyfish | Release Notes | August 11, 2022 | April 2027 | April 2032 |
| Ubuntu **22.04 LTS** | Jammy Jellyfish | Release Notes | April 21, 2022 | April 2027 | April 2032 |
| Ubuntu **20.04.5 LTS** | Focal Fossa | Changes | September 1, 2022 | April 2025 | April 2030 |
| Ubuntu **20.04.4 LTS** | Focal Fossa | Changes | February 24, 2022 | April 2025 | April 2030 |
| Ubuntu 20.04.3 LTS | Focal Fossa | Changes | August 26, 2021 | April 2025 | April 2030 |
| Ubuntu 20.04.2 LTS | Focal Fossa | Changes | February 4, 2021 | April 2025 | April 2030 |
| Ubuntu 20.04.1 LTS | Focal Fossa | Changes | August 6, 2020 | April 2025 | April 2030 |
| Ubuntu 20.04 LTS | Focal Fossa | Release Notes | April 23, 2020 | April 2025 | April 2030 |
| Ubuntu **18.04.6 LTS** | Bionic Beaver | Changes | September 17.2021 | June 2023 | April 2028 |
| Ubuntu 18.04.5 LTS | Bionic Beaver | Changes | August 13, 2020 | June 2023 | April 2028 |
| Ubuntu 18.04.4 LTS | Bionic Beaver | Changes | February 12, 2020 | June 2023 | April 2028 |
| Ubuntu 18.04.3 LTS | Bionic Beaver | Changes | August 8, 2019 | June 2023 | April 2028 |
| Ubuntu 18.04.2 LTS | Bionic Beaver | Changes | February 15, 2019 | June 2023 | April 2028 |
| Ubuntu 18.04.1 LTS | Bionic Beaver | Changes | July 26, 2018 | June 2023 | April 2028 |
| Ubuntu 18.04 LTS | Bionic Beaver | Release Notes | April 26, 2018 | June 2023 | April 2028 |

https://wiki.ubuntu.com/Releases


Ubuntu 22.04 LTS (Jammy Jellyfish)
releases.ubuntu.com/22.04/

# Select an image

Ubuntu is distributed on two types of images described below.

## Desktop image

The desktop image allows you to try Ubuntu without changing your computer at all, and at your option to install it permanently later. This type of image is what most people will want to use. You will need at least 2048MiB of RAM to install from this image.

64-bit PC (AMD64) desktop image

Choose this if you have a computer based on the AMD64 or EM64T architecture (e.g., Athlon64, Opteron, EM64T Xeon, Core 2). Choose this if you are at all unsure.

https://releases.ubuntu.com/22.04/ubuntu-22.04-desktop-amd64.iso

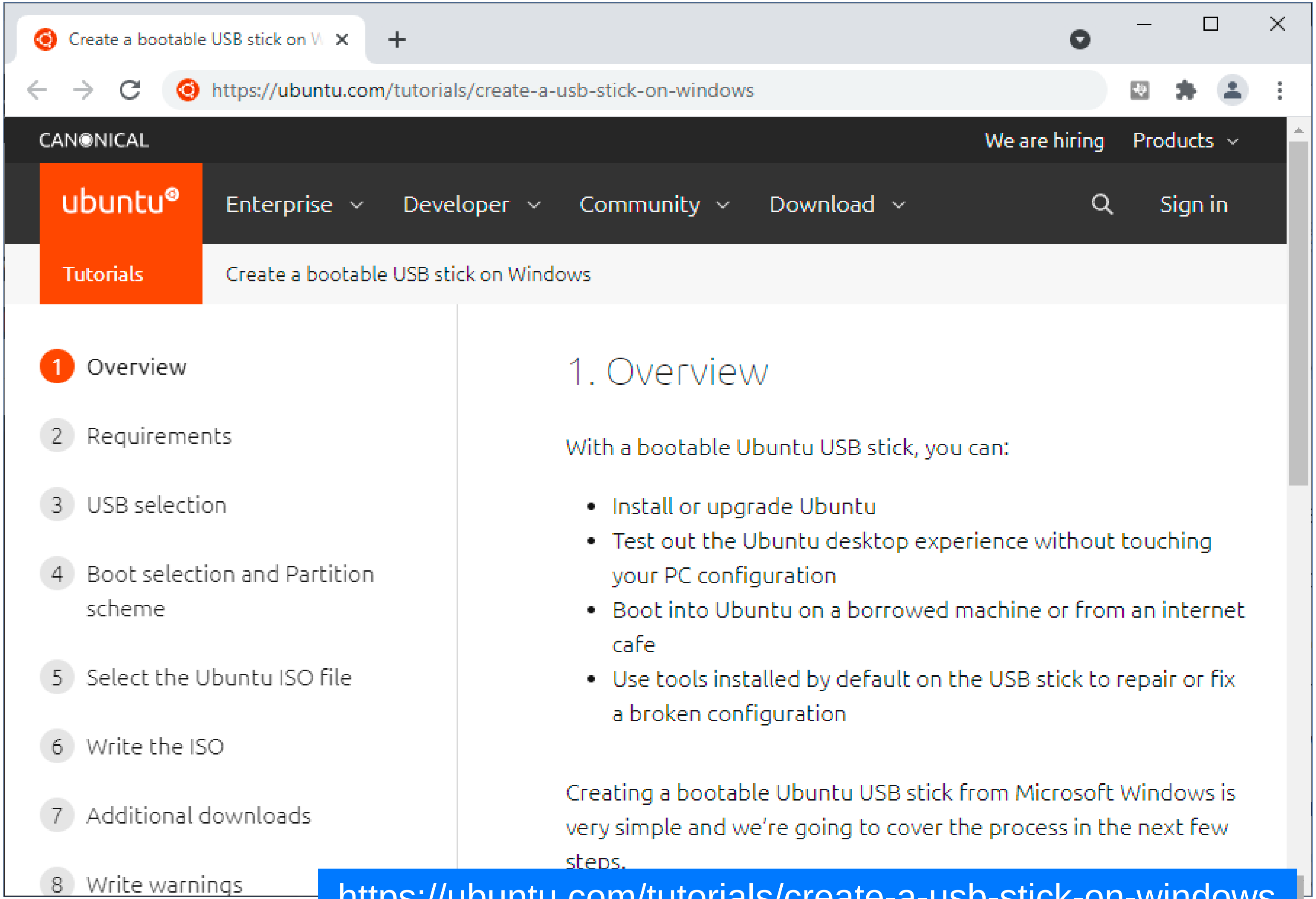
## Server install image

The server install image allows you to install Ubuntu permanently on a computer for use as a server. It will not install a graphical user interface.

64-bit PC (AMD64) server install image

Choose this if you have a computer based on the AMD64 or EM64T architecture (e.g., Athlon64, Opteron, EM64T Xeon, Core 2). Choose this if you are at all unsure.

https://releases.ubuntu.com/22.04/ubuntu-22.04-live-server-amd64.iso



Create a bootable USB stick on W
https://ubuntu.com/tutorials/create-a-usb-stick-on-windows
CANONICAL
We are hiring
Products
ubuntu
Enterprise
Developer
Community
Download
Sign in
Tutorials
Create a bootable USB stick on Windows
1 Overview
2 Requirements
3 USB selection
4 Boot selection and Partition scheme
5 Select the Ubuntu ISO file
6 Write the ISO
7 Additional downloads
8 Write warnings

## 1. Overview

With a bootable Ubuntu USB stick, you can:

- Install or upgrade Ubuntu
- Test out the Ubuntu desktop experience without touching your PC configuration
- Boot into Ubuntu on a borrowed machine or from an internet cafe
- Use tools installed by default on the USB stick to repair or fix a broken configuration

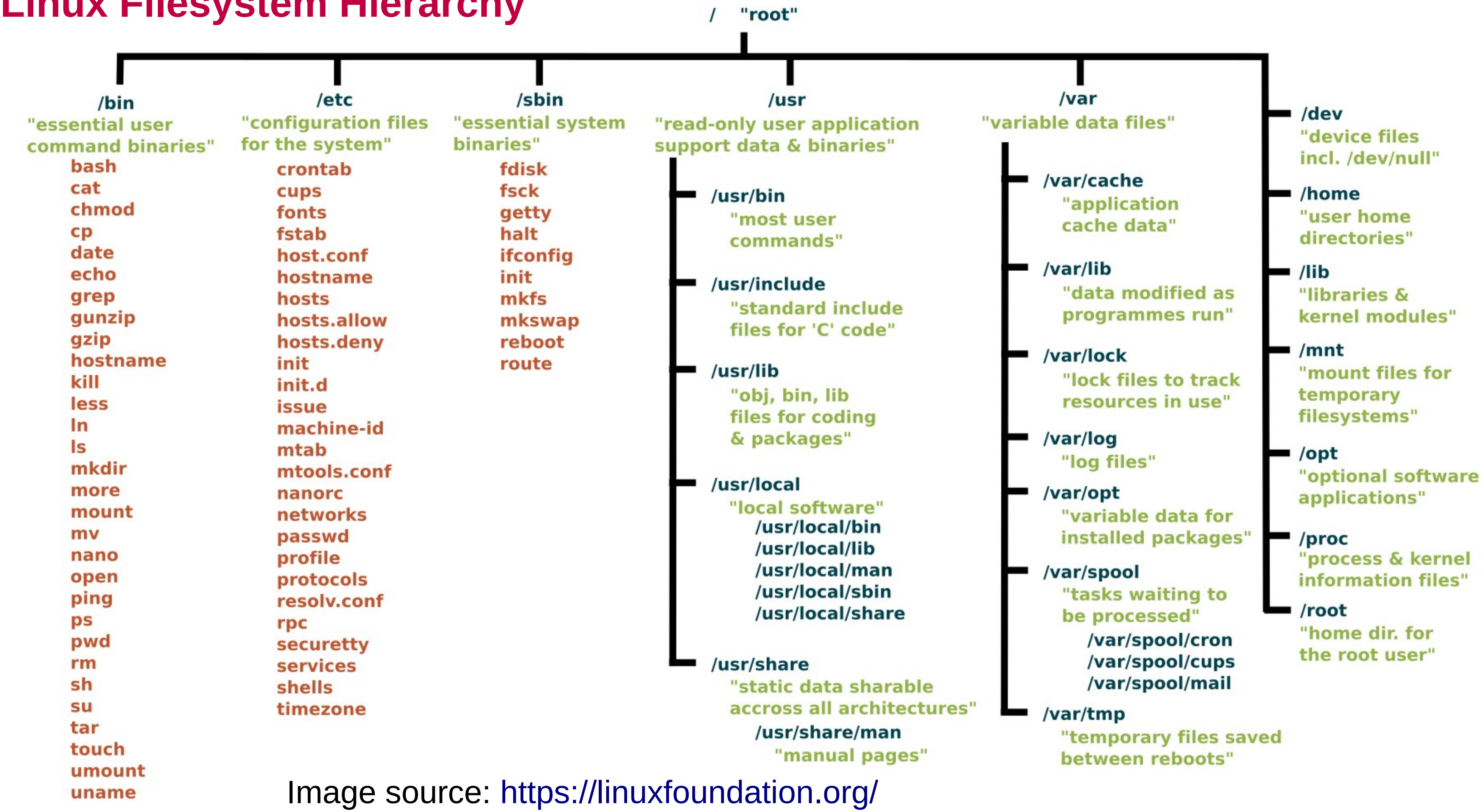
Creating a bootable Ubuntu USB stick from Microsoft Windows is very simple and we're going to cover the process in the next few steps.

https://ubuntu.com/tutorials/create-a-usb-stick-on-windows

# Some Basic Linux Commands

```
awk
cat
cd
chmod
chown
clear
cp
curl
date
df
du
find
free
grep
groupadd
groupdel
gzip
head
history
ifconfig
ip
kill
less
ls
mkdir
more
mount
mv
passwd
ping
ps
pwd
reboot
rm
rmdir
scp
sed
shutdown
ssh
su
tail
tar
top
touch
umount
uname
unzip
useradd
userdel
usermod
wc
wget
whoami
```

# Linux Filesystem Hierarchy

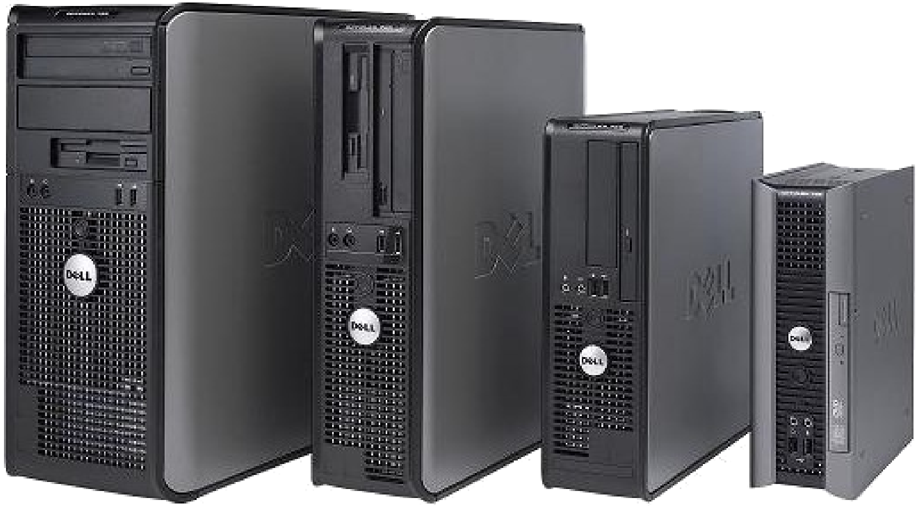


/ "root"

- /bin "essential user command binaries"
  - bash
  - cat
  - chmod
  - cp
  - date
  - echo
  - grep
  - gunzip
  - gzip
  - hostname
  - kill
  - less
  - ln
  - ls
  - mkdir
  - more
  - mount
  - mv
  - nano
  - open
  - ping
  - ps
  - pwd
  - rm
  - sh
  - su
  - tar
  - touch
  - umount
  - uname
- /etc "configuration files for the system"
  - crontab
  - cups
  - fonts
  - fstab
  - host.conf
  - hostname
  - hosts
  - hosts.allow
  - hosts.deny
  - init
  - init.d
  - issue
  - machine-id
  - mtab
  - mtools.conf
  - nanorc
  - networks
  - passwd
  - profile
  - protocols
  - resolv.conf
  - rpc
  - securetty
  - services
  - shells
  - timezone
- /sbin "essential system binaries"
  - fdisk
  - fsck
  - getty
  - halt
  - ifconfig
  - init
  - mkfs
  - mkswap
  - reboot
  - route
- /usr "read-only user application support data & binaries"
  - /usr/bin "most user commands"
  - /usr/include "standard include files for 'C' code"
  - /usr/lib "obj, bin, lib files for coding & packages"
  - /usr/local "local software"
    - /usr/local/bin
    - /usr/local/lib
    - /usr/local/man
    - /usr/local/sbin
    - /usr/local/share
  - /usr/share "static data sharable accross all architectures"
    - /usr/share/man "manual pages"
- /var "variable data files"
  - /var/cache "application cache data"
  - /var/lib "data modified as programmes run"
  - /var/lock "lock files to track resources in use"
  - /var/log "log files"
  - /var/opt "variable data for installed packages"
  - /var/spool "tasks waiting to be processed"
    - /var/spool/cron
    - /var/spool/cups
    - /var/spool/mail
  - /var/tmp "temporary files saved between reboots"
- /dev "device files incl. /dev/null"
- /home "user home directories"
- /lib "libraries & kernel modules"
- /mnt "mount files for temporary filesystems"
- /opt "optional software applications"
- /proc "process & kernel information files"
- /root "home dir. for the root user"

Image source: https://linuxfoundation.org/

# Different Types of Computers (Physical Hardware)



Mini Tower

Dekstop

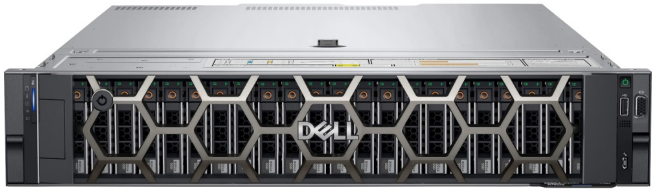
Small Form Factor

Ultra Form Factor

Notebook

Tablet

# Different Types of Computers (Physical Hardware)



Rack-mountable Servers

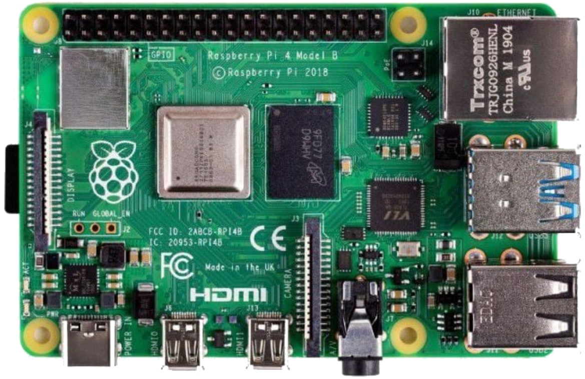
Image Source: DELL

# Different Types of Computers (Physical Hardware)

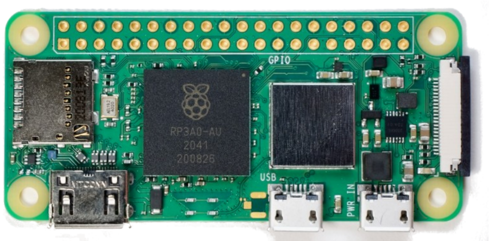
DELL
Mini Desktop
Micro PC

Intel NUC "Nook"
(Ultra-small PC)

Raspberry Pi 4
Model B



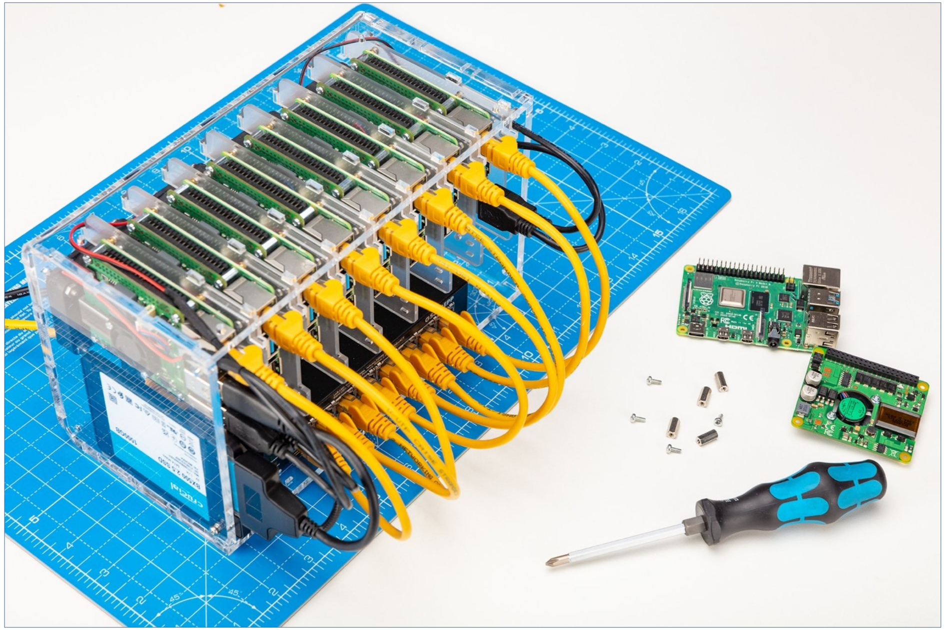
Raspberry Pi
Zero 2 W



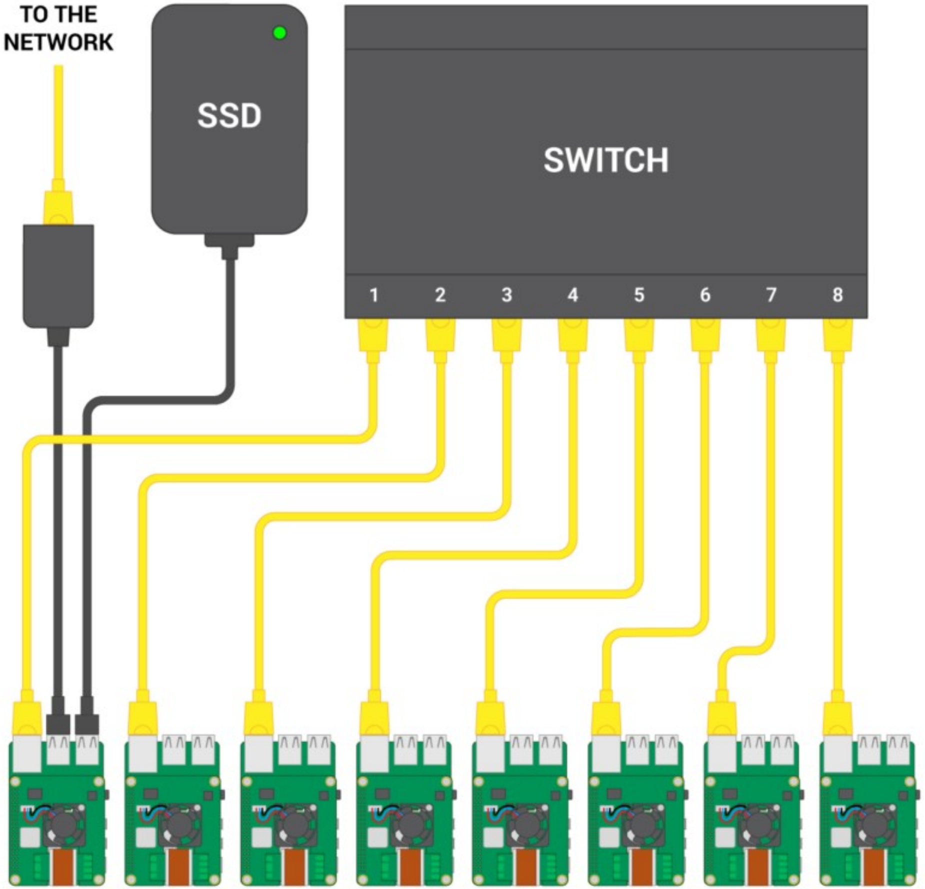
# Raspberry Pi Clusters

External 1TB SSD mounted via a USB3-SATA connector

Gigabit Ethernet-USB3 Ethernet dongle



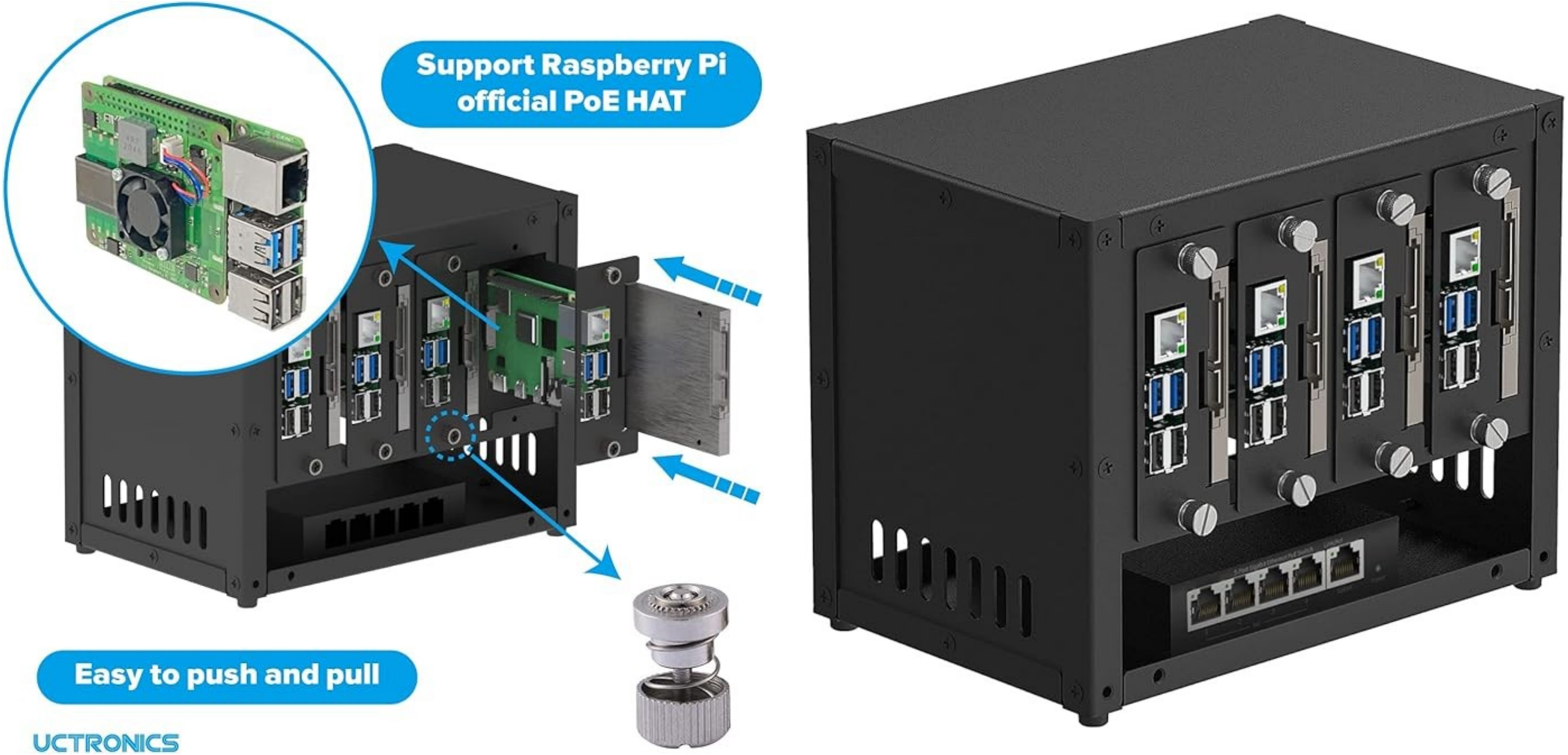
Network switch with PoE ports



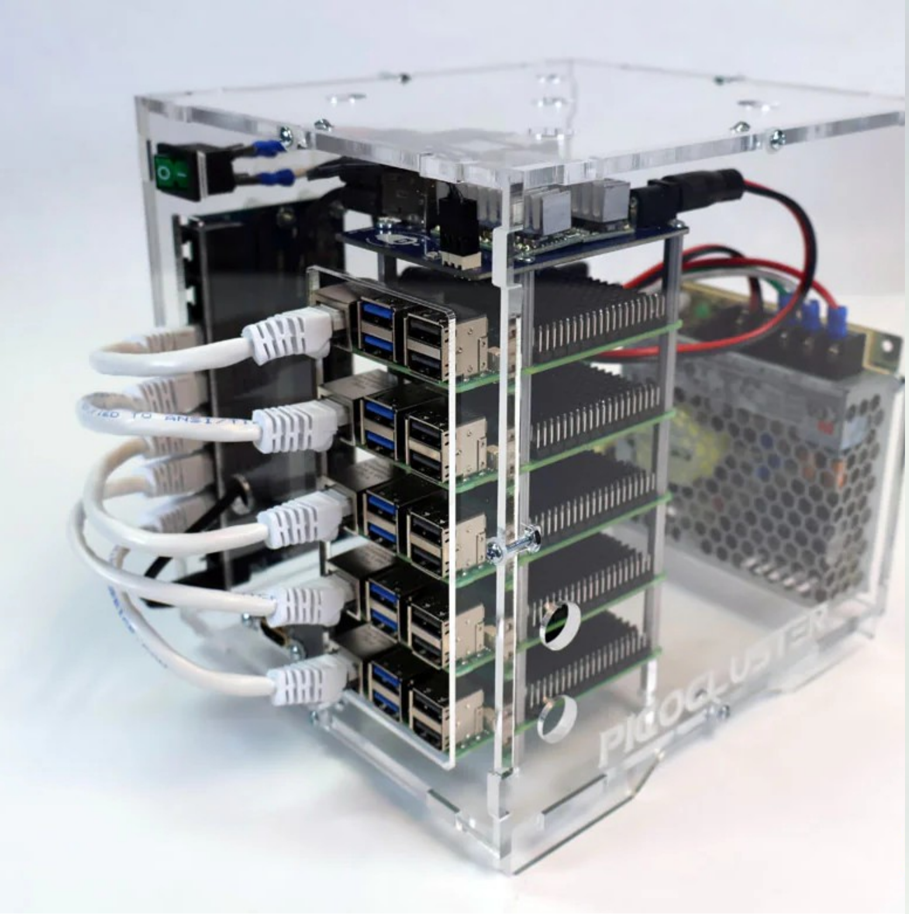
Head node

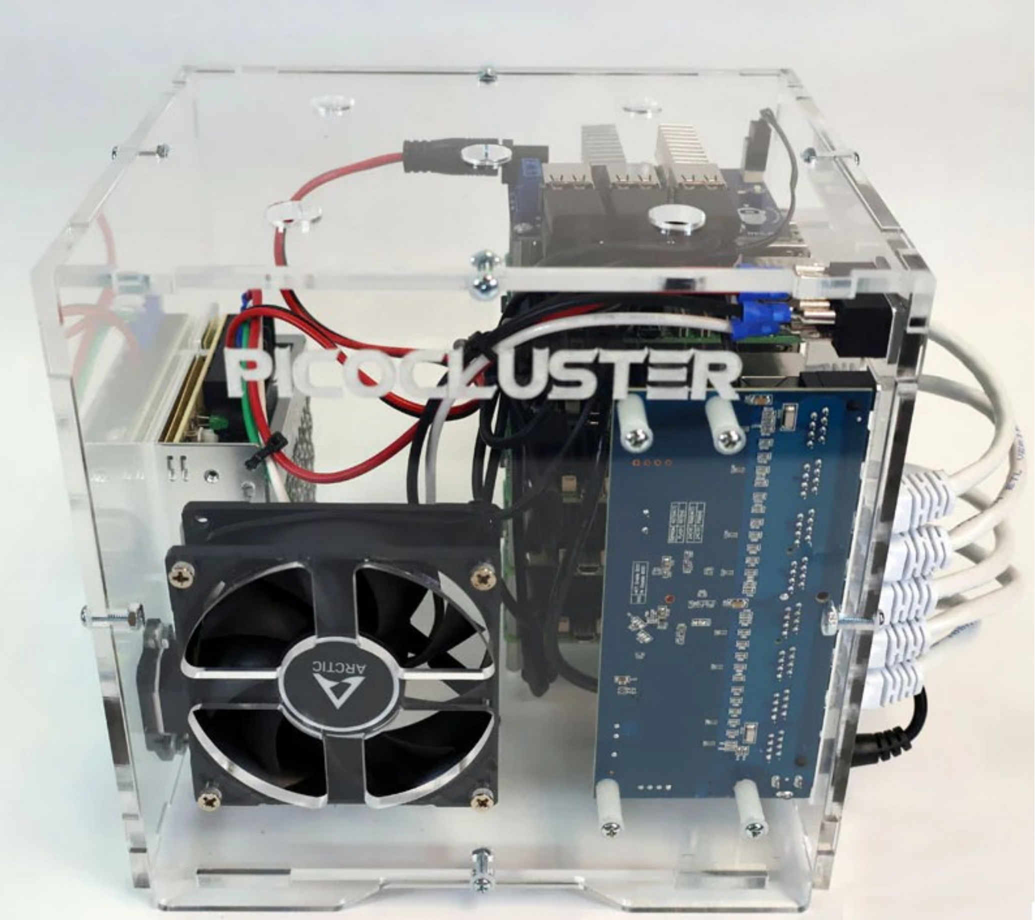
Raspberry Pi + PoE+ HAT network boot

https://www.raspberrypi.com/tutorials/cluster-raspberry-pi-tutorial/


Support Raspberry Pi official PoE HAT
Easy to push and pull
UCTRONICS

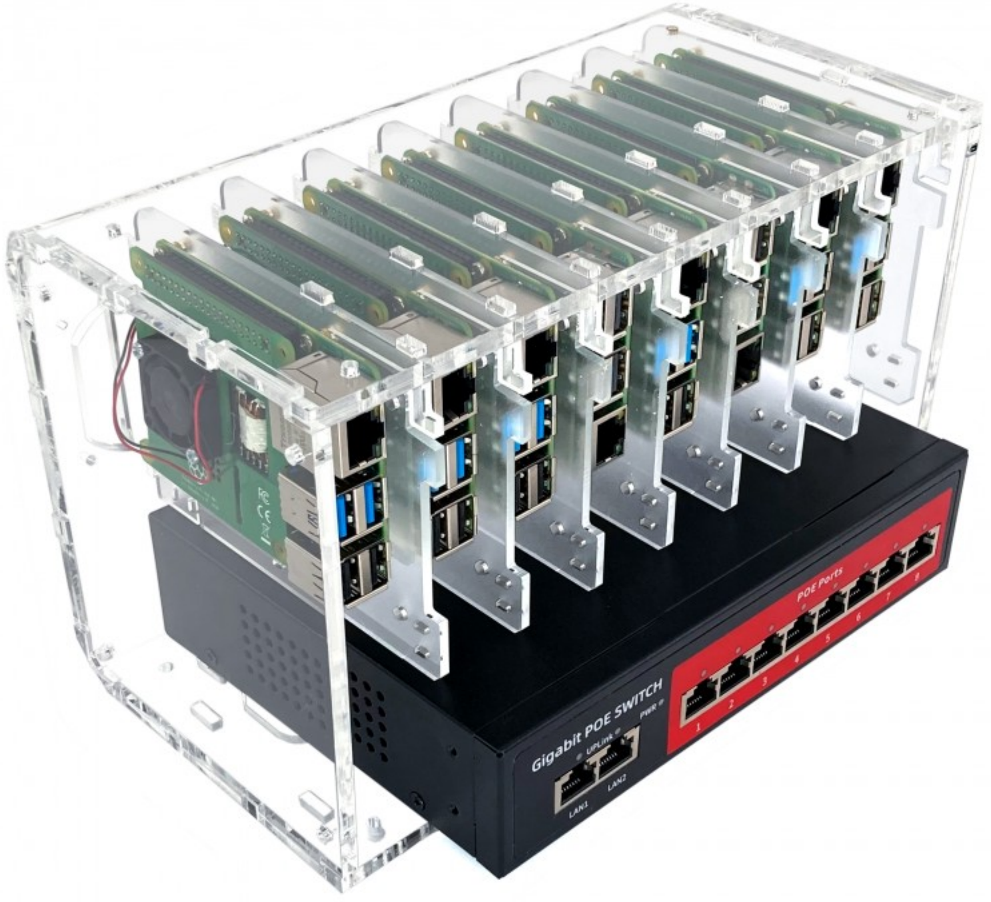
# Pico 5H Raspberry PI5 Cluster





https://www.picocluster.com/products/pico-5-raspberry-pi5-cluster-8gb

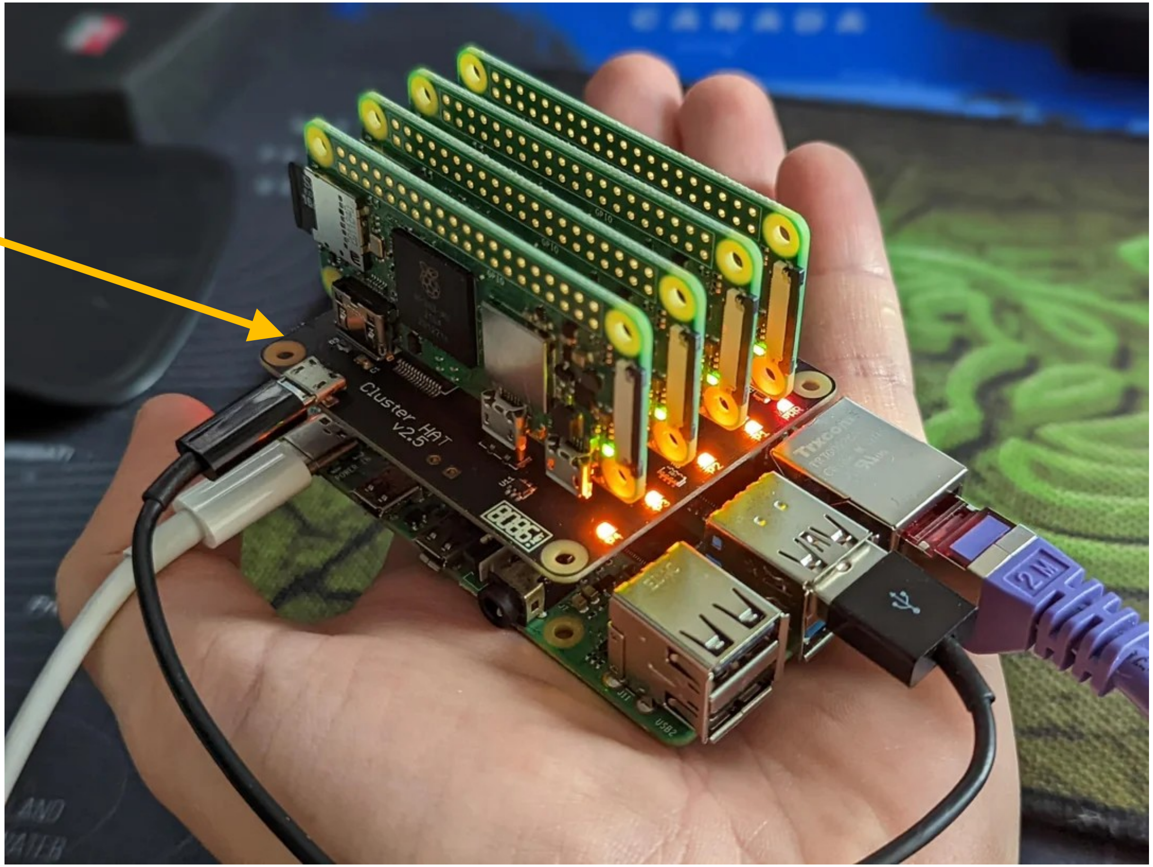
# 8 Slot Cloudlet Cluster Case



https://www.c4labs.com/product/cloudlet-cluster-case-raspberry-pi/

# A tiny cluster based on 4x Raspberry Pi Zero 2 W

Cluster HAT



# Hardware Virtualization

- Hardware virtualization is the creation of **multiple virtual machines (VMs)** that can run different operating systems and applications on a single **shared physical computer**.
- A layer of software called a **hypervisor (virtualization software)** is installed on the **host machine** and used to manage the VMs.
- Each VM is allocated a portion of the host machine's resources, including CPU, memory, storage, and network resources.

# Types of Hypervisors

- A **type-1 (bare-metal) hypervisor** runs directly on the host machine and provides direct access to the host's resources, such as CPU, memory, and storage, to the virtual machines.
- This allows for high performance and minimal overhead, as the VMs have direct access to the hardware.

# Types of Hypervisors

- A **type-2 (hosted) hypervisor** runs on top of a host OS, such as Windows or Linux, and provides a layer of abstraction between the virtual machines and the host's hardware.
- This can result in higher overhead and reduced performance compared to Type 1 hypervisors.

# Types of Hypervisors

- Examples of **Type 1 hypervisors**:
  - VMware ESXi
  - Microsoft Hyper-V
  - KVM (built into Linux OS)
- Examples of **Type 2 hypervisors**:
  - Oracle VirtualBox
  - VMware Workstation

# Linux Virtual Machines

- Here are some advantages of using a Virtual Machine for learning Linux.
  - Running a Linux VM ensures complete isolation from the host operating system. As a result, any modifications made to the Linux system will not have an impact on the host system.
  - Using a Linux VM enables one to conduct experiments with diverse Linux versions and configurations without the concern of damaging their primary system.

# Oracle VirtualBox

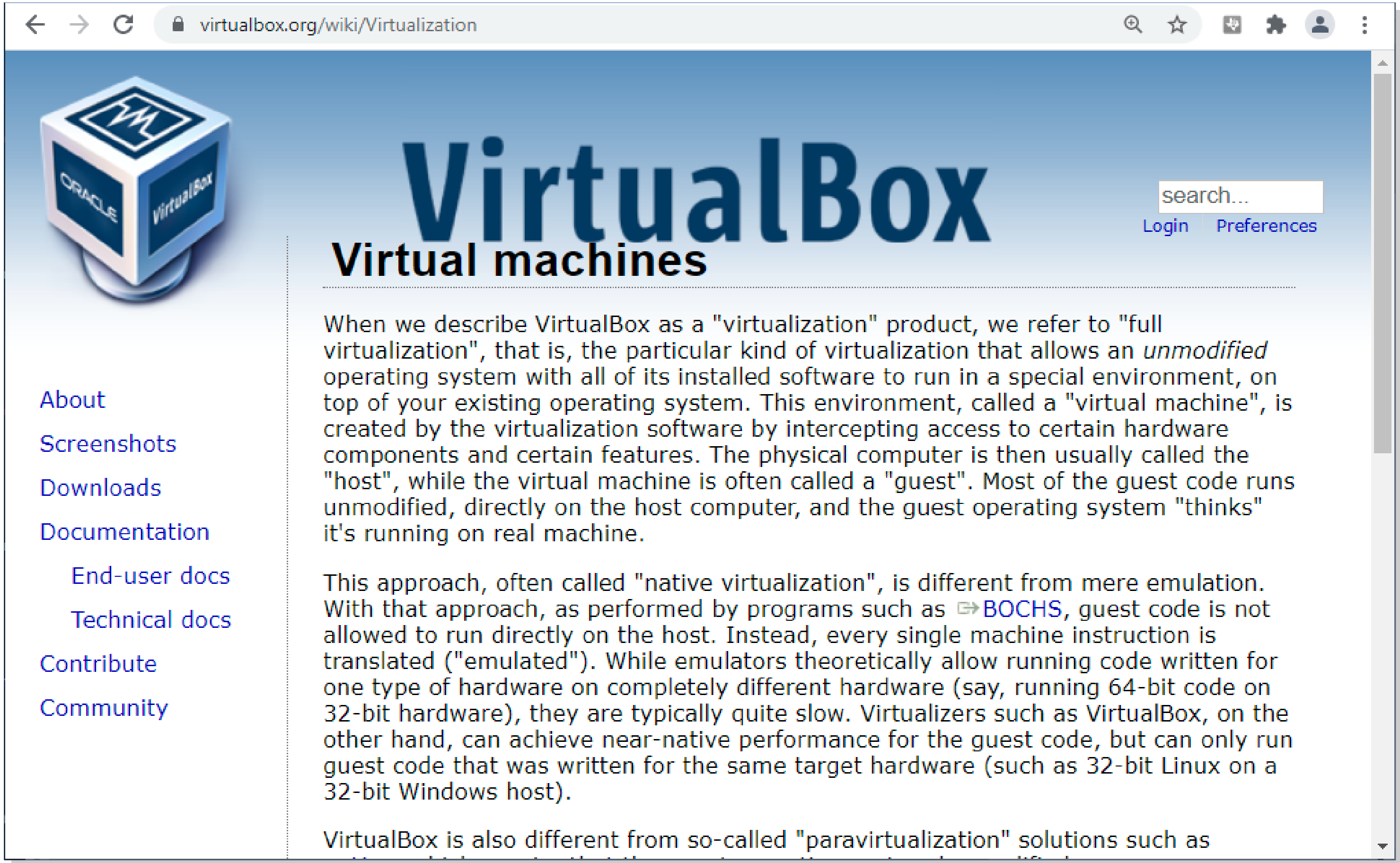
- Oracle VirtualBox is a free and **open-source virtualization software** (**type-2 hypervisor**).
- It enables users to create VMs that can run a wide range of guest operating systems, including Windows, Linux and macOS.
- This allows users to test software on different OSes, run legacy applications, and isolate potentially dangerous programs.

https://www.virtualbox.org/

# Oracle VirtualBox

- VirtualBox provides support for VM snapshots, which allow users to save the state of a VM at a specific moment and restore it later.
- It also supports advanced networking options, such as **virtual LANs** and **virtual network adapters**, which can be useful for testing network configurations and applications.

**VirtualBox** is a virtualization platform for x86 and AMD64 / Intel64.



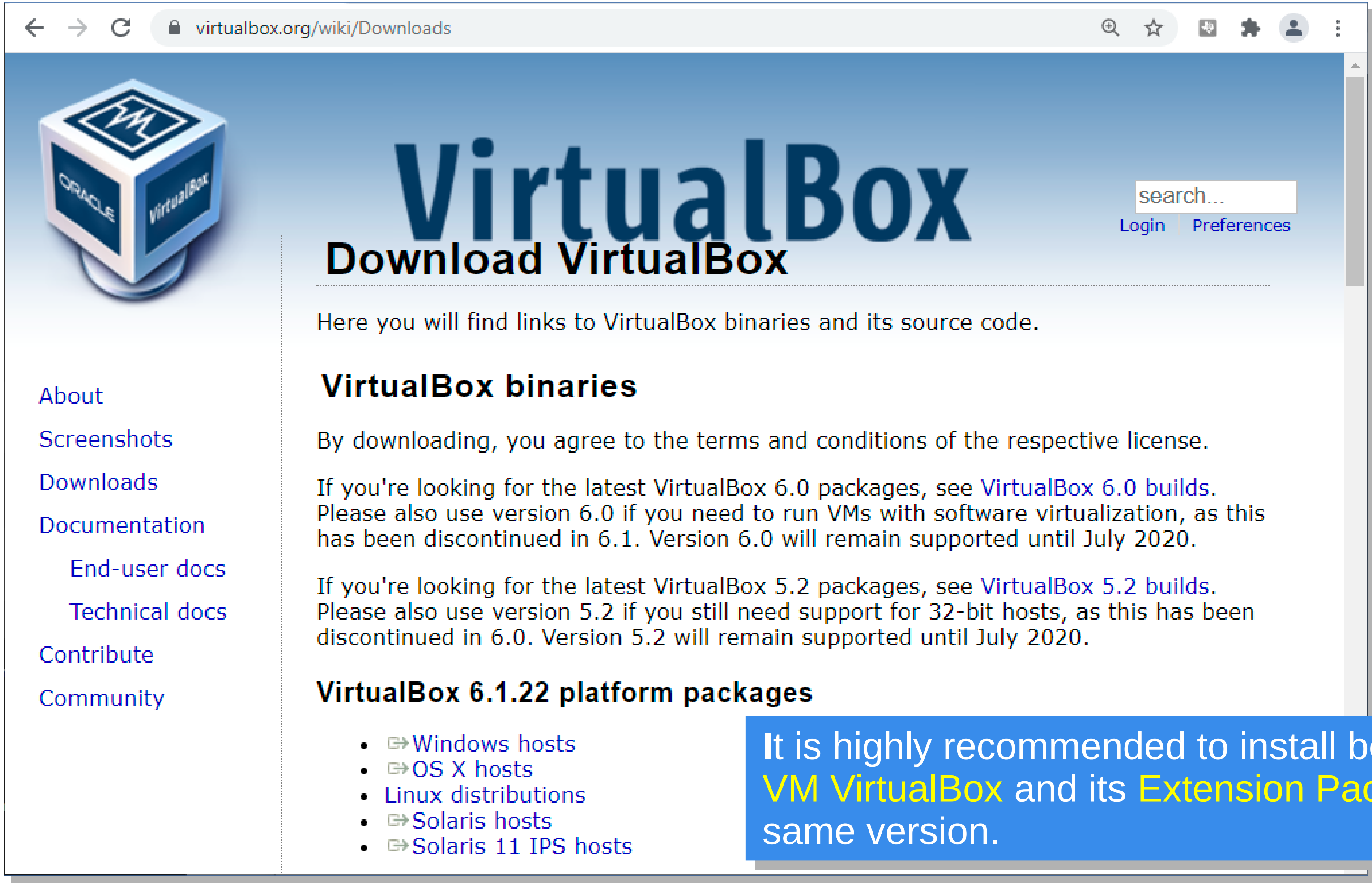
## Virtual machines

When we describe VirtualBox as a "virtualization" product, we refer to "full virtualization", that is, the particular kind of virtualization that allows an *unmodified* operating system with all of its installed software to run in a special environment, on top of your existing operating system. This environment, called a "virtual machine", is created by the virtualization software by intercepting access to certain hardware components and certain features. The physical computer is then usually called the "host", while the virtual machine is often called a "guest". Most of the guest code runs unmodified, directly on the host computer, and the guest operating system "thinks" it's running on real machine.

This approach, often called "native virtualization", is different from mere emulation. With that approach, as performed by programs such as BOCHS, guest code is not allowed to run directly on the host. Instead, every single machine instruction is translated ("emulated"). While emulators theoretically allow running code written for one type of hardware on completely different hardware (say, running 64-bit code on 32-bit hardware), they are typically quite slow. Virtualizers such as VirtualBox, on the other hand, can achieve near-native performance for the guest code, but can only run guest code that was written for the same target hardware (such as 32-bit Linux on a 32-bit Windows host).

VirtualBox is also different from so-called "paravirtualization" solutions such as

https://www.virtualbox.org/wiki/Downloads



# Download VirtualBox

Here you will find links to VirtualBox binaries and its source code.

## VirtualBox binaries

By downloading, you agree to the terms and conditions of the respective license.

If you're looking for the latest VirtualBox 6.0 packages, see VirtualBox 6.0 builds. Please also use version 6.0 if you need to run VMs with software virtualization, as this has been discontinued in 6.1. Version 6.0 will remain supported until July 2020.

If you're looking for the latest VirtualBox 5.2 packages, see VirtualBox 5.2 builds. Please also use version 5.2 if you still need support for 32-bit hosts, as this has been discontinued in 6.0. Version 5.2 will remain supported until July 2020.

### VirtualBox 6.1.22 platform packages

- ⇨Windows hosts
- ⇨OS X hosts
- Linux distributions
- ⇨Solaris hosts
- ⇨Solaris 11 IPS hosts

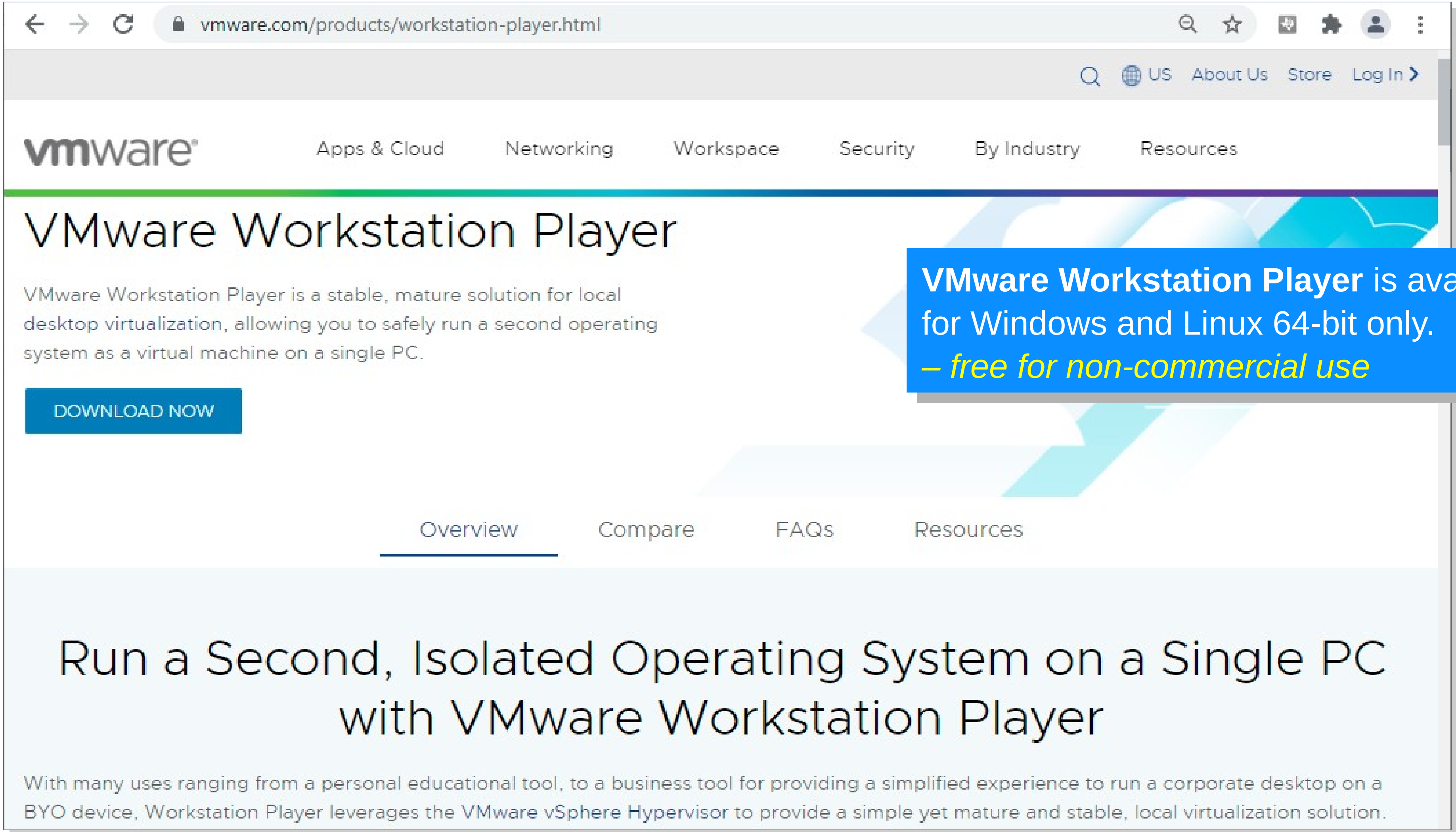
It is highly recommended to install both Oracle VM VirtualBox and its Extension Pack of the same version.

https://www.virtualbox.org/wiki/Downloads

# VMware Workstation Player

- VMware Workstation Player is a desktop virtualization software.
- It is a simplified version of **VMware Workstation Pro**, which is a more feature-rich and advanced version of the software.
- It is **free for personal, non-commercial use**.
  - However, for commercial use, a license is required.

https://www.vmware.com/products/workstation-player.html



VMware Workstation Player

VMware Workstation Player is a stable, mature solution for local desktop virtualization, allowing you to safely run a second operating system as a virtual machine on a single PC.

Run a Second, Isolated Operating System on a Single PC with VMware Workstation Player

With many uses ranging from a personal educational tool, to a business tool for providing a simplified experience to run a corporate desktop on a BYO device, Workstation Player leverages the VMware vSphere Hypervisor to provide a simple yet mature and stable, local virtualization solution.

**VMware Workstation Player** is available for Windows and Linux 64-bit only.
*– free for non-commercial use*

# KVM

- **KVM (Kernel-based Virtual Machine)** is open source software that converts Linux into a type-1 (bare-metal) hypervisor.
- **KVM** can only be used on a processor with **hardware virtualization extensions** such as **Intel-VT** or **AMD-V**.

https://ubuntu.com/blog/kvm-hyphervisor

# QEMU (Quick Emulator)

- QEMU is an **open source machine emulator and virtualizer**. For example, it can be used to boot an Linux OS image for an ARM-based processor board.
- It also supports a variety of hardware emulation options, including CPUs, network devices, and storage controllers.

https://www.qemu.org/

# Options for installing / using Linux

**1)** Install **Linux** on a bootable **USB Flash Drive with capacity of 8GB (or larger)** and boot the computer from the flash drive as the first choice (changing BIOS settings may be necessary).

**2)** Install **WSL2 software** and a **Linux distro** on a desktop computer running **Microsoft Windows 10 or 11**.

**3)** Install a **Linux** distro **side-by-side with an existing OS** on a separate partition of a hard disk.
→ **Multi-boot environment**.

# Options for installing / using Linux

**4)** Install a **Hypervisor** or **Virtual Machine (VM) player** on a desktop computer (running **Windows** as the **host OS**) and install a Linux distro as a **guest OS**.

**5)** Install a Linux distro on a **VPS (Virtual Private Server)** provided by a **VPS hosting or Cloud provider** (with extra usage cost).

**6)** Install a Linux distro on a **single-board computer (SBC)** such as **Raspberry Pi** (with hardware cost).

# Options for Installing Ubuntu VMs

- **VMware Workstation Player** *(free for non-commercial use)*
- **Oracle Virtual Box** *(free)*
- **Ubuntu Multipass** *(free)*

# Ubuntu Multipass

- Multipass is a tool for creating Ubuntu VMs.
  - recommended method for creating Ubuntu VMs on Ubuntu.
- Multipass installs on Linux, Windows and macOS.
- Multipass uses **KVM** on Linux, **Hyper-V** on Windows and **HyperKit** on macOS to run VMs.

https://ubuntu.com/server/docs/virtualization-multipass

https://multipass.run/

# WSL

- WSL stands for “***Windows Subsystem for Linux***”.
- It is a virtualization technology for computers running Microsoft Windows, including Windows 10 and 11.
- It allows Windows users to run a Linux operating system environment directly on Windows.

# WSL

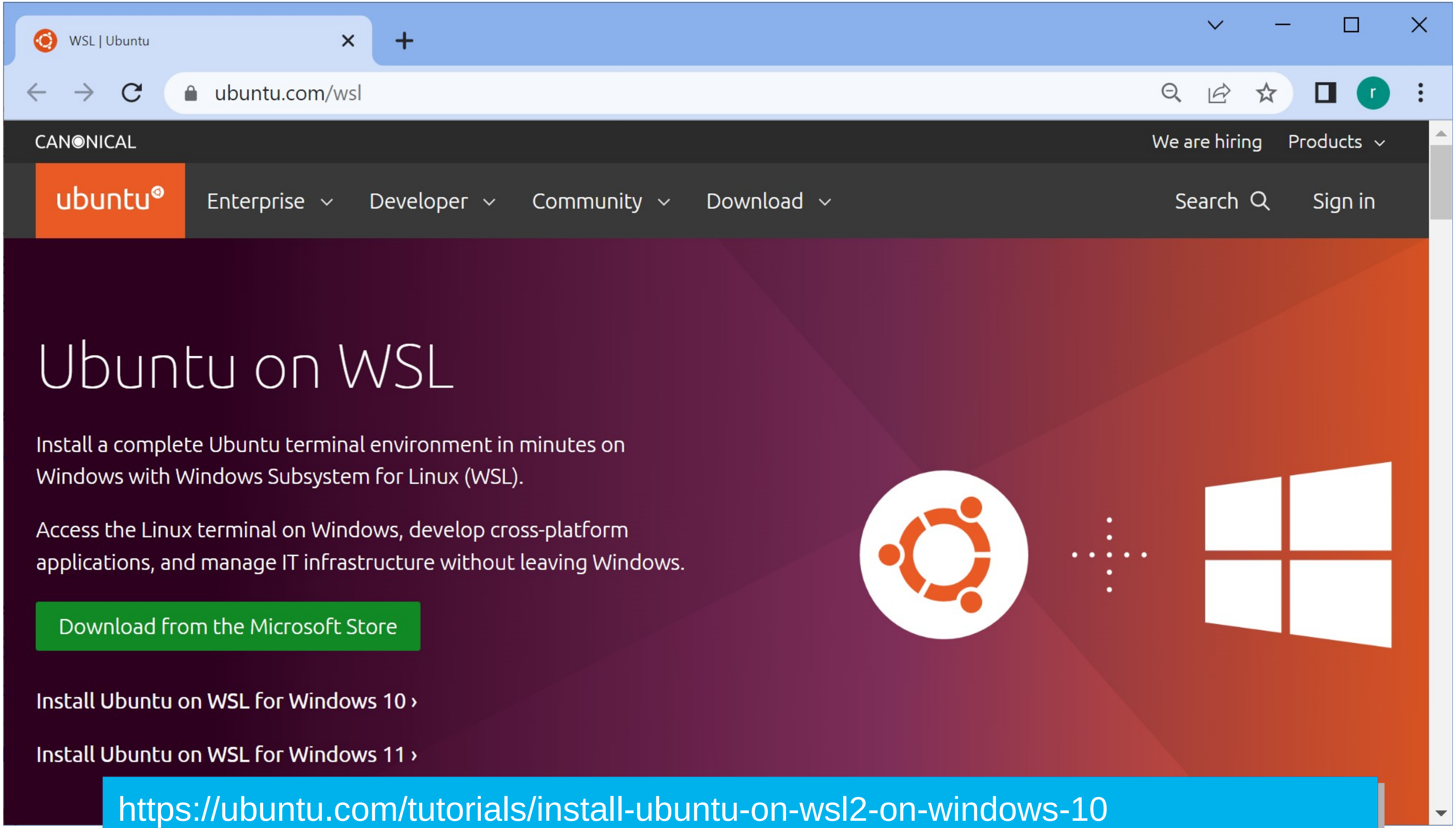
- **WSL2** is an improvement over the previous version (WSL version 1), offering enhanced performance and compatibility with Linux software.
- **WSL2** utilizes the Microsoft Hyper-V hypervisor to run a lightweight VM that hosts both the Linux kernel and user-mode components.

# WSL

- **WSL2** allows Linux applications to run natively on Windows, with access to the same file system and hardware resources as Windows applications.
- **WSL2** also includes improved support for running **Docker containers on Windows**, which can be useful for software development and testing.

# WSL Online Documentation

- **Windows Subsystem for Linux Documentation**
  - https://learn.microsoft.com/en-us/windows/wsl/
- **Install Linux on Windows with WSL**
  - https://learn.microsoft.com/en-us/windows/wsl/install
- **Basic commands for WSL**
  - https://learn.microsoft.com/en-us/windows/wsl/basic-commands



# Ubuntu on WSL

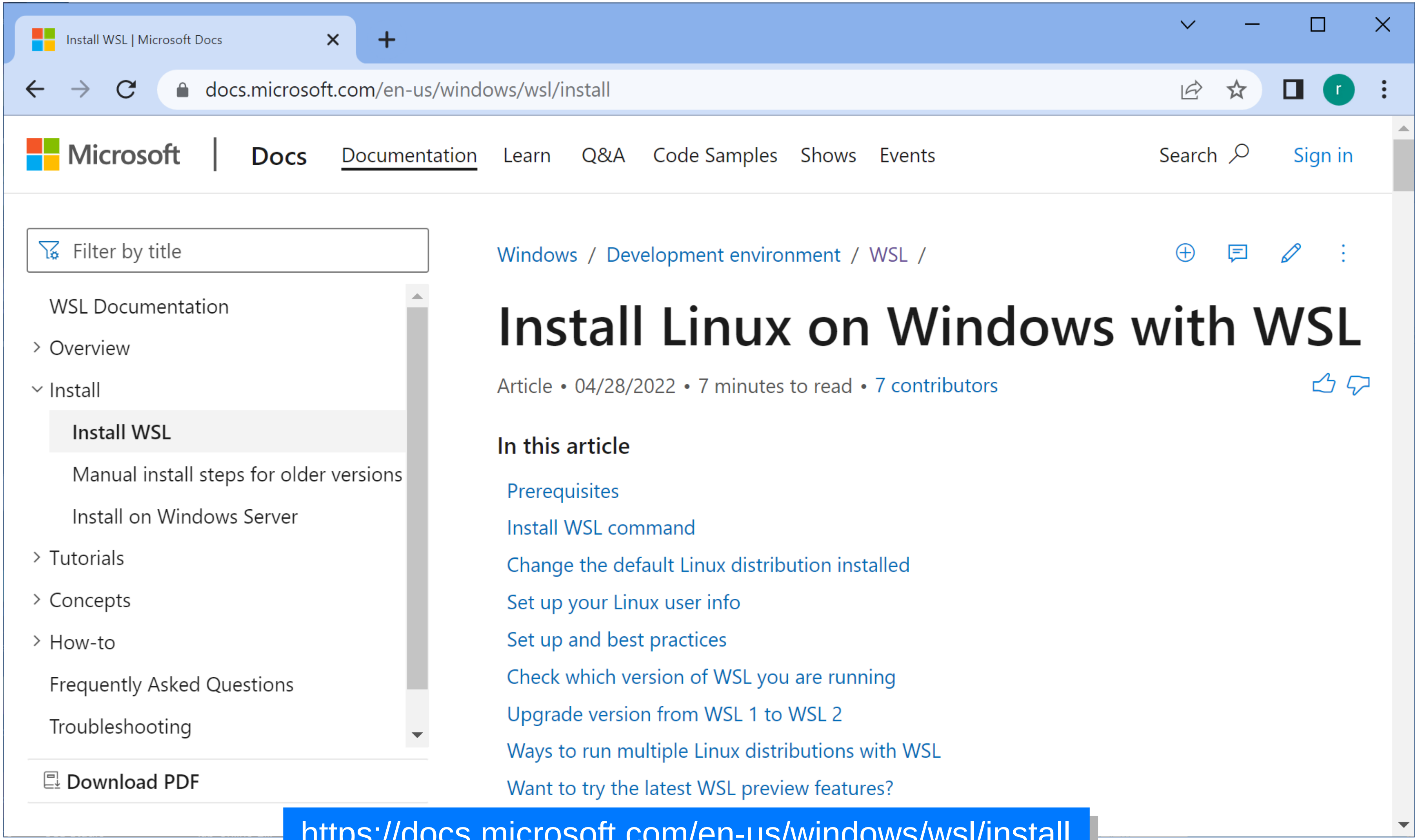
Install a complete Ubuntu terminal environment in minutes on Windows with Windows Subsystem for Linux (WSL).

Access the Linux terminal on Windows, develop cross-platform applications, and manage IT infrastructure without leaving Windows.

Install Ubuntu on WSL for Windows 10 ›

Install Ubuntu on WSL for Windows 11 ›

https://ubuntu.com/tutorials/install-ubuntu-on-wsl2-on-windows-10
https://ubuntu.com/tutorials/install-ubuntu-on-wsl2-on-windows-11-with-gui-support



Install WSL | Microsoft Docs
docs.microsoft.com/en-us/windows/wsl/install
Microsoft
Docs
Documentation
Learn
Q&A
Code Samples
Shows
Events
Search
Sign in
Filter by title
WSL Documentation
Overview
Install
Install WSL
Manual install steps for older versions
Install on Windows Server
Tutorials
Concepts
How-to
Frequently Asked Questions
Troubleshooting
Download PDF
Windows / Development environment / WSL /

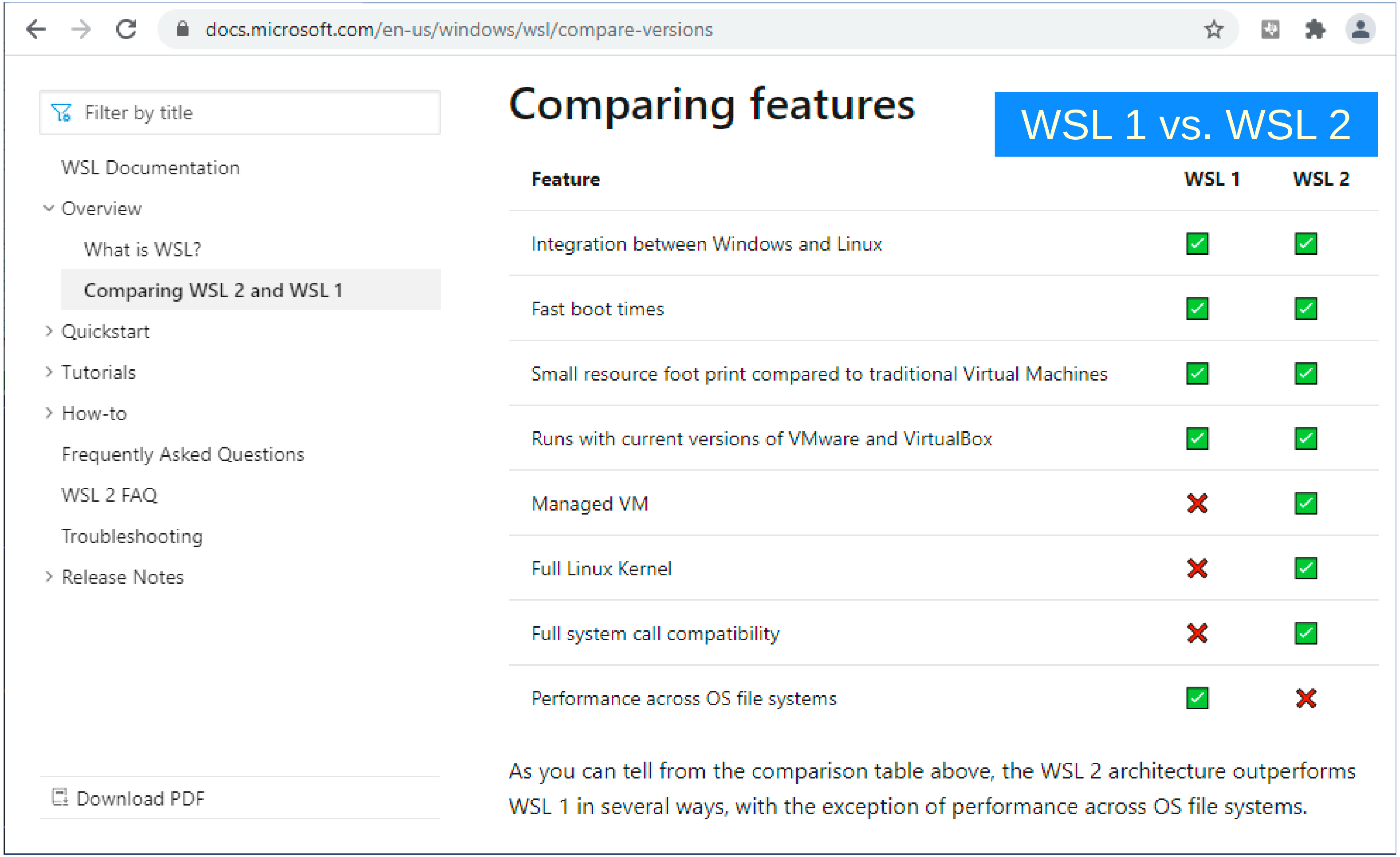
# Install Linux on Windows with WSL

Article • 04/28/2022 • 7 minutes to read • 7 contributors

## In this article

- Prerequisites
- Install WSL command
- Change the default Linux distribution installed
- Set up your Linux user info
- Set up and best practices
- Check which version of WSL you are running
- Upgrade version from WSL 1 to WSL 2
- Ways to run multiple Linux distributions with WSL
- Want to try the latest WSL preview features?

https://docs.microsoft.com/en-us/windows/wsl/install



## Comparing features

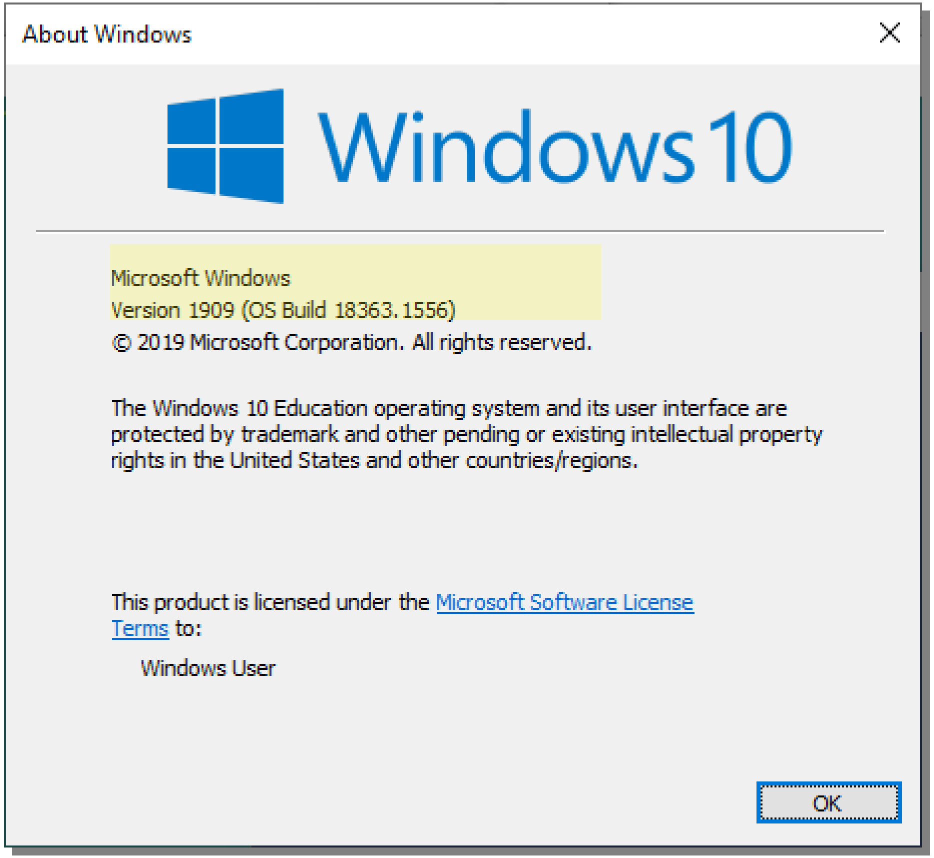
WSL 1 vs. WSL 2

| Feature | WSL 1 | WSL 2 |
|---|---|---|
| Integration between Windows and Linux | ✓ | ✓ |
| Fast boot times | ✓ | ✓ |
| Small resource foot print compared to traditional Virtual Machines | ✓ | ✓ |
| Runs with current versions of VMware and VirtualBox | ✓ | ✓ |
| Managed VM | ✗ | ✓ |
| Full Linux Kernel | ✗ | ✓ |
| Full system call compatibility | ✗ | ✓ |
| Performance across OS file systems | ✓ | ✗ |

As you can tell from the comparison table above, the WSL 2 architecture outperforms WSL 1 in several ways, with the exception of performance across OS file systems.

https://learn.microsoft.com/en-us/windows/wsl/compare-versions

For **Windows 10** users, check the OS version with the command '`winver`' (run the command in a PowerShell terminal) before installing WSL2.
– **WSL2** is only available in **Windows Build 18362 or higher**.

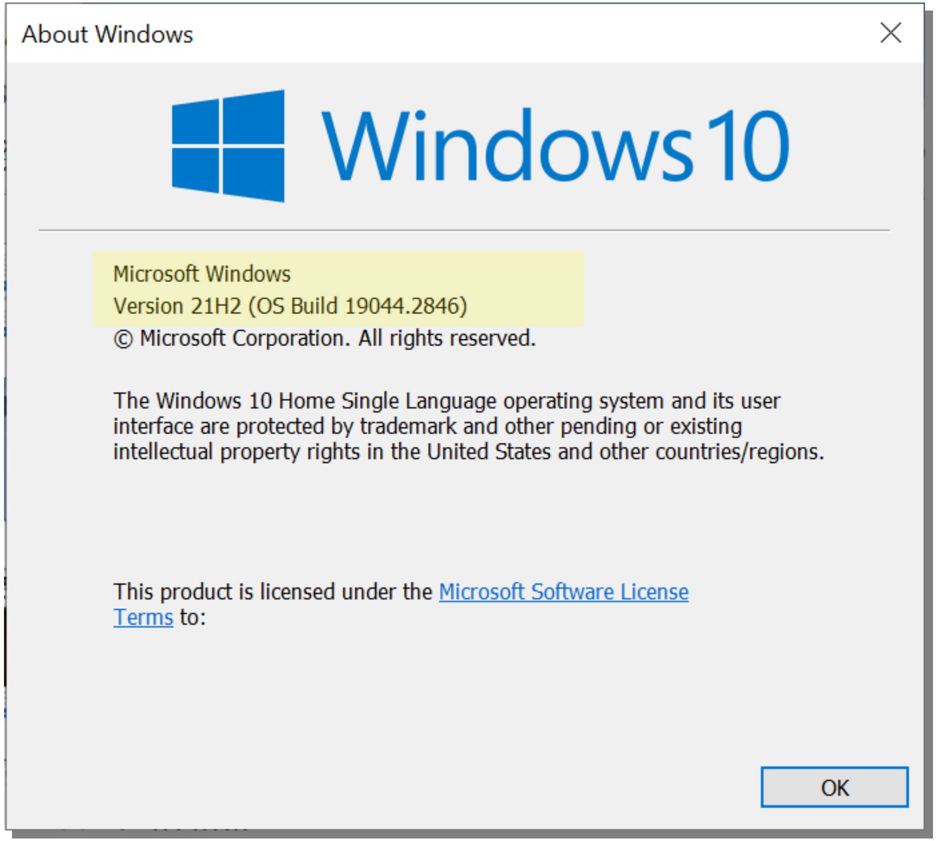

About Windows

Microsoft Windows
Version 1909 (OS Build 18363.1556)
© 2019 Microsoft Corporation. All rights reserved.

The Windows 10 Education operating system and its user interface are protected by trademark and other pending or existing intellectual property rights in the United States and other countries/regions.

This product is licensed under the Microsoft Software License Terms to:

Windows User

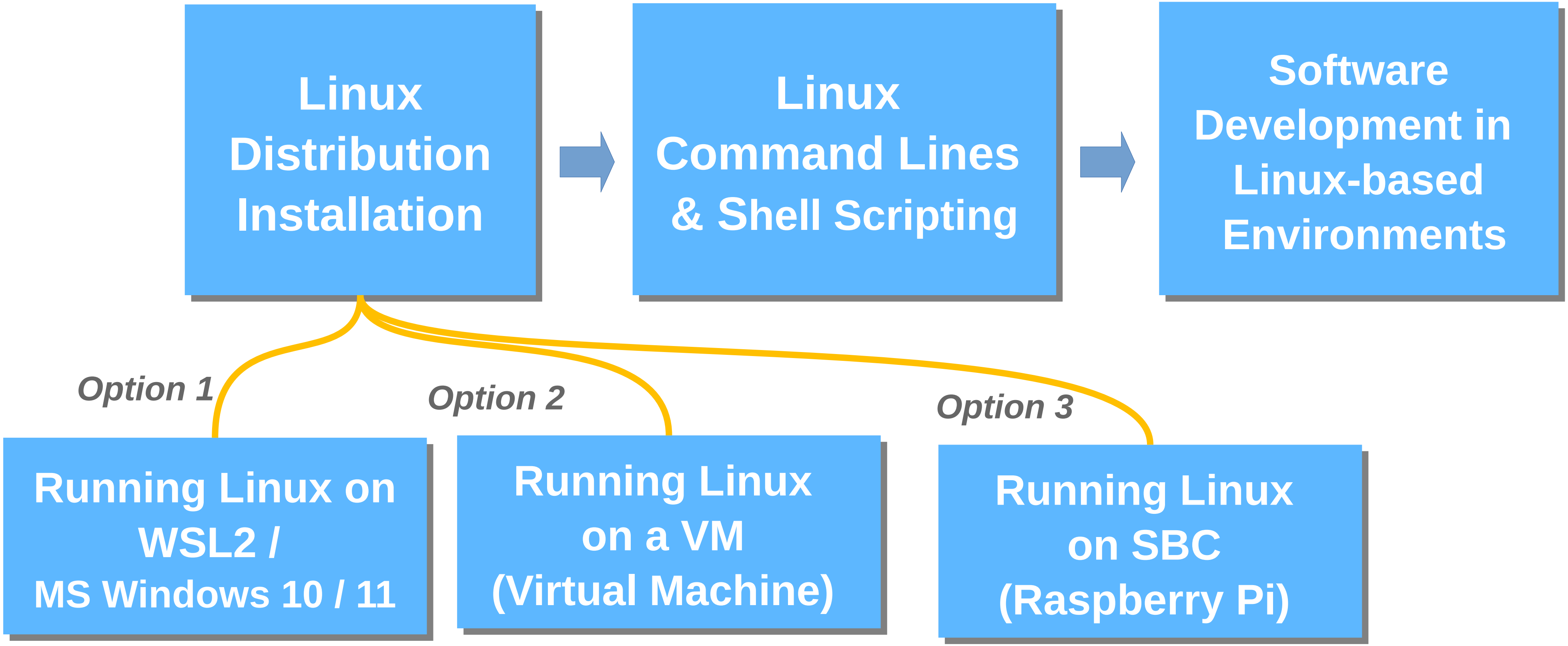

About Windows

Microsoft Windows
Version 21H2 (OS Build 19044.2846)
© Microsoft Corporation. All rights reserved.

The Windows 10 Home Single Language operating system and its user interface are protected by trademark and other pending or existing intellectual property rights in the United States and other countries/regions.

This product is licensed under the Microsoft Software License Terms to:

# Linux-based Environment for Software Development



Linux Distribution Installation

Linux Command Lines & Shell Scripting

Software Development in Linux-based Environments

*Option 1*

Running Linux on WSL2 / MS Windows 10 / 11

*Option 2*

Running Linux on a VM (Virtual Machine)

*Option 3*

Running Linux on SBC (Raspberry Pi)

*for Windows users*