

# Picking an Operating System



How to choose the right OS or Distro for your Home Lab

# What is an Operating System?

- Once a computer has booted it has to load a more general purpose environment to get work done in
- BIOS is the way a computer initially boots and then hands off to an OS
- Operating Systems are usually comprised of a kernel that is immediately loaded, an initialization environment that is loaded after the kernel and then some libraries and services that are available to use
- There are a wide variety of different flavors of operating systems that have been developed over the years

# A brief history of operating systems

...

# Families of Operating Systems

## Common

- Mainframe (50s)
- UNIX (70s)
- DOS/Windows (80s)
- Mac OS (80s)
- Linux (90s)
- BSD (90s)
- Windows NT (90s)
- macOS (00s)

## Uncommon

- BeOS (90s)
- Plan 9 (90s)
- AmigaOS (80s)
- OS/2 (80s)
- CP/M (70s)
- VAX/VMS (70s)

# UNIX

- Initially developed at AT&T Bell Labs starting in 1969
- Mainframe operating systems were too complex and there was a revolution in available smaller minicomputers in the market
- Driven to deliver new innovative features such as multi-tasking and portability
- Heavily influenced by academics and was widely adopted by Universities
- Initially well adopted due to permissive initial license by academic institutions
- Now a standard held by the The Open Group an industry standards consortium

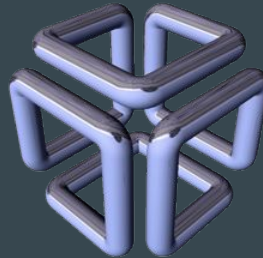


# License Agreements and the birth of Open Source

- AT&T licensed UNIX to a universities and corporations because it was forbidden to enter the computer business from an antitrust case
- When AT&T was broken up Bell Laboratories started selling UNIX commercially
- This ultimately led to fragmentation in the ecosystem with a number of UNIX variants but no real standard for compilers and runtimes
- Many people wanted to see their contributions not be swallowed by a corporation
- Proponents of open source created new operating systems and user land software that were licensed with these new open source license agreements
- This led to further fragmentation initially of the community but ultimately more choice for low or no cost operating systems and software available

# UNIX Licenses

- IBM - AIX
- Sun Microsystems - SunOS, Solaris
  - OpenSolaris, illumos, OmniOS
- Silicon Graphics International - IRIX
- Hewlett-Packard Company - HP-UX
- Digital Equipment Corporation - Digital Unix, Tru64



# Personal Computer Revolution

- IBM Personal Computer (1981) and Microsoft DOS were capitalizing further on the trend miniaturization of computers
- Apple would introduce Apple I/II and then the Lisa/Macintosh in response and up the ante with a new graphical user interface
- Companies like Atari, Commodore and others were trying to bleed the line of game console and computer





# Linux

- Linux is a monolithic operating system kernel
  - It has two parts a kernel and an initial ram disk (initrd)
- Created by Linus Torvalds in 1991
- Licensed with the GNU General Public License version 2
  - Derivative work must be licensed under the same terms
- GNU/Linux will be used to describe the general family of operating systems with the linux kernel with the GNU userland (glibc, bash, etc.)
- Distribution is the flavor GNU/Linux with often the most important differentiation being its package manager
- Maybe the most influential software ever written?

# BSD

- Berkeley Standard Distribution
- University of California, Berkeley was an early licensee of the original UNIX and started creating their own distribution (1978)
- Starting in 1984 the license agreements for AT&T were expensive so they proposed to remove and re-write the AT&T sections of BSD.
- The result was ported to the now almost ubiquitous Intel 80386 platform and became the basis of the FreeBSD, NetBSD and OpenBSD
- BSD was a pioneer in the implementation of networking protocols like TCP/IP
- After the port these were licensed under the permissive free license BSD licenses
  - This requires the copyright notice for derived software but not full source to be provided

# Choosing a Distribution

...

# GNU/Linux Distribution Breeds

IBM Red Hat was one of the first well packaged distros for Linux in the early 90s. They an upstream model where the bleeding edge distros are support for short periods (6-12 months) but are the starting points the basis for their longer term enterprise commercial products.

- Red Hat Enterprise Linux
  - CentOS (rolling)
  - Alma Linux (rolling)
  - Rocky (ABI)
- Fedora Linux

Debian is a non-commercial organization that has a manifesto that an operating system should be maintained in the spirit of Linux and GNU. Distro provides a very wide number of architecture ports. Commercial support is available but only on distributions that downstream.

- Debian
  - Stable
  - Testing
  - Unstable
- Ubuntu
  - Bi-yearly release with some LTS (5 year)

# GNU/Linux Distribution Choices

- Package Manager
  - APT
  - YUM/DNF
  - Others (Nix, pacman, etc.)
- Init (Process 1)
  - Systemd
  - SysV init
- From scratch
  - Linux From Scratch
  - NixOS
  - Gentoo
  - Etc.



**Red Hat**  
Enterprise Linux



debian



archlinux



fedora



Ubuntu



NixOS



Rocky Linux™

# GNU/Linux Package Managers

## Advanced package tool (APT)

- Found on Debian based distros
- These will be the most common commands found when searching the internet
- Uses the dpkg package format
- More complicated packaging format with limited documentation due to not being provided by a commercial company

## Dandified YUM (Yellowdog Updater Modifier)

- Found on Red Hat based distros
- Uses the rpm package format
- Relatively straightforward package format that users can easily create new packages with good documentation

# GNU/Linux Init

- Systemd is now the defacto standard found in all Red Hat based distributions and Debian/Ubuntu distributions.
- There is a smaller but strong community that does not agree with this decision
- Systemd detractors believe it violates the UNIX philosophy of make each program do one thing well. That the kitchen sink approach leads to software bloat, complexity and a much larger attack surface.
- Unless you are using a fringe GNU/Linux operating system or one that builds from scratch you may not have a choice
- Systemd does make things like service dependencies and resource control much easier for system administrators than prior options

# So you want to build GNU/Linux yourself

Probably don't at least not initially.

Scratch distributions are really good learning tools but they often have pitfalls. Home Labs will benefit from having the resources to experiment and learn from these. However if you have production activities these will most likely be fragile and brittle.

Rolling back and managing updates can also be very challenging and in the current world of software exploits it is often important to be able to update quickly to a patched release of software (and know that your system is impacted).



# Modern BSD

Home Lab environments are very much network environments which can require software routers and firewalls. Some of the strongest use cases for BSD are in these scenarios with their very strong networking core. They also due to more permissive licenses can include encumbered software like ZFS for best of breed file systems.

The limitation is these are generally only going to be available x86 hardware currently.

- pfSense/OPNsense
  - Router and firewall appliance distributions which have very strong functionality for doing network address translation, virtual private networking and firewalling.
- TrueNAS
  - Open-core based on FreeBSD that provides a storage appliance that can provide block, file and object storage to your network

# Purpose built GNU/Linux

Much like some of the BSD distributions there are some very interesting and useful GNU/Linux distributions that are useful to investigate.

- **OpenWrt**
  - Open-source embedded system distributions for network routing including specifically for open source wireless access points
- **Proxmox Virtual Environment**
  - Open-source server virtualization platform for Kernel-based Virtual Machines (KVM)

**Club Lab operating system  
choices?**

...

# Not every lab we prep can target every possible operating system.

- If we did labs with all completely different machines. We would either have to check them all for gotchas or we'd lose a lot of time debugging issues that are only holding up one or two people.
- There are so many edge cases that any instructions would get bogged down in them and miss out on the interesting parts.

# Goals for picking a target operating system

- We want something that club members will be able to find good supporting documentation
- Needs to work on the platform that we have initially chosen for hardware

# Raspberry Pi OS

- Originally called Raspbian
- As name implies it is a Debian derived operating system
- Obviously going to have very good compatibility with the hardware
- While mature Home Labs will have multiple operating systems or distros potentially in them this is going to be the lowest cost entry point for most people to get started
- Many Linux distributions are similar so skills learned on one will translate to others relatively easily