

# Maintaining a Linux Distribution

Leah Neukirchen <leah@vuxu.org>

Practical Course: Contributing to an Open-Source Project  
November 17, 2020

# Why are Linux distributions?

---

- Unlike most other operating systems, a usable Linux system needs to be assembled from multiple projects:
  - The Linux kernel
  - GNU userland
  - X.org or Wayland
  - Desktops such as KDE/Gnome/XFCE...
  - ...many many other tools and packages
- with many possible variations and decisions in details  
→ many possible distributions!

# Differences between Linux distributions

---

- target audience: newbies, advanced users, artists, security researchers, ham radio operators...
- source vs. binary packages
- release model: periodic, sporadic, rolling release
- platform support
  
- package manager

# Differences between Linux distributions

---

- target audience: newbies, **advanced users**, artists, security researchers, ham radio operators...
- source vs. **binary packages**
- release model: periodic, sporadic, **rolling release**
- platform support: **i686, x86\_64, armv6, armv7, aarch64, powerpc...**
- package manager: **XBPS**
- **Example: Void Linux**

# A too short history of Void Linux

---

2008 created from scratch by Juan Romero Pardines as a testbed for XBPS

2014 move to runit, adapting my init system ignite (built for Arch Linux)

2014-07-02 my first commit

2015 I actually switch to Void on my main machine

2020 21 team members,  
148 people had 5 or more commits in 2020

# What does a maintainer do?

---

- Keeping packages up to date:
  - Void's target is providing the latest stable release of software
  - I maintain over 580 packages and whatever else comes up
  - Daily effort: usually less than 30 minutes (0-5 packages)
    - (but then occasionally you spend three nights updating Perl)

# What does a maintainer do?

---

- Internal communication:
  - design decisions
  - work delegation
  - investigating bug reports
  - writing announcements and documentation
- Communication with other projects:
  - forwarding bug reports and patches to upstream
  - importing patches from other distributions
  - monitoring security announces

# What does a maintainer do?

---

- Packaging new software:
  - figuring out build systems and dependencies
  - integrating the package into the system (e.g. into the init system)
  - enforcing policies (paths, usernames, optional features, ...)

# Politics

---

- Void is a rather small project and many things are decided ad-hoc. (Counterexample: Debian.)
- Flat management hierarchy that decides on new members.
- Do-ocracy? You can commit anything, but don't break what you can't fix.
- Openness: You can contribute easily without being a member.

# Lessons learned: automation

---

- How can a small group of people keep up with such an effort? → Use tools to automate tedious tasks!
- We wrote tools to:
  - check for new package releases
  - find and report common mistakes in contributions
  - automatically update packages that need to be rebuilt
  - work fully CI/CD driven: updating a package and releasing it just takes a Git commit
- Contributions are tested on third-party CI systems (formerly Travis, now GitHub Actions)
- Packages can be contributed via Pull Requests, and merged with one click → enabling drive-by contributions, a good way to find new maintainers

# Lessons learned: distributing power

---

- the initial founder of the distribution was missing from 2018–2020
  - we could keep working pretty well, but there were single points of control:
    - domain names
    - control over the GitHub project
  - we “forked” the project before Juan came back in 2020
    - new domains
    - move to a new GitHub organization
    - unavoidable breakage of some things
- all access is shared now to at least two persons
- we do not have own funds, eschewed that problem

# Lessons learned?

## Maintainers are humans

---

- over time, new people join the project!
- but also, people leave
  - burnout
  - personal problems
  - conflicts inside the project?
  - other things become more interesting
- compared to other projects, distro work never stops
- finding a balance can be hard
- take care of yourself and others

**Questions?**

**Thank you.**