Taking control over the means of production: Free software boot

Denis 'GNUtoo' Carikli

March 19, 2023

Controlling the means of production is the path to freedom

Real examples of taking control of the production:

- The peasent movements / Via campesina
- Anarchist revolution in 1936 in Spain
- Zapatistas, Temporary autonomous zone, etc
- Theorized through work on the commons by Elinor Ostrom, Sivia Federici (Feminism and the politics of the commons), and probably many others people as well.

The free software way

- Considered as a commons but problematic dependencies (make few hardware, we don't make our own chips yet)
- "Reverse engineering, re-implementation and parallel operation": Hacking as transgressive infrastructuring:
  https://web.archive.org/web/20230000000000*/https:
  //mkorn.binaervarianz.de/pub/korn-cscw2016.pdf
- (Re)implementations: From applications like Emacs, GCC (GNU) to the OS (GNU, GNU/Linux) and to free software Boot, hardware freedom, etc.
- (Also goes into other domains, but not the focus of this presentation).

Introduction
000

From free software applications down to the hardware
●000000000000000

Using the certifications
00000000000

Under the hood and DIY approach
0000000000000000000000000

ARM computers with u-boot
000000000000

Different levels of freedom

Free applications $\rightarrow$ Common distro + common hardware $\rightarrow$ RYF + FSDG $\rightarrow$ Hardware freedom

Introduction
000

From free software applications down to the hardware
0●0000000000000000

Using the certifications
00000000000

Under the hood and DIY approach
0000000000000000000000000

ARM computers with u-boot
000000000000

Free software applications

Use case: You already run GNU/Linux and want to do collective activities

- Communication application: Example: gpg + email client
- Organization (association, company): Example: Libreoffice

What if people don't already run free software OS?

OS	Free compiler?	Restrictions?
Microsoft Windows	Yes (Mingw64, Msys2, Guix)	signed drivers (Wireshark and ncap)
Mac OS	No	Complicated to run (signed applications)
Android	Difficult / limited	usually no root access
iOS (iphone / ipads)	No	Apple prevents runing free software applications, censors applications, forbids real browsers, etc

Issues

- The applications have no control over the hardware: they need to ask the OS which then access network, the display etc on behalf of applications:
    - A nonfree OS could spy on our mails for instance (Exemple: Microsoft adding plugins inside Firefox that had at least one security issue) → Puts people at risk and big issue witin movements that don't have the same agenda as Microsoft or states. https://www.computerworld.com/article/2763687/sneaky-microsoft-plug-in-puts-firefox-users-at-risk.html
    - The OS can refuse to run an application or free software applications (iphones / ipads), prevent access to some resources (network drivers), etc.
- Lot more issues, just an example: [insert your own issues here].
- → The OS must be free software

Solution: help people upgrade from nonfree OS to GNU/Linux

- Install parties
- Finding people interested (might require to go find them outside of your community)

Introduction
From free software applications down to the hardware
Using the certifications
Under the hood and DIY approach
ARM computers with u-boot

Extremely important but not sufficient (has many limitations)

The FSF has certifications for distributions and hardware: What can go wrong without them?

Introduction
000

From free software applications down to the hardware
000000000●00000000

Using the certifications
000000000000

Under the hood and DIY approach
0000000000000000000000000

ARM computers with u-boot
000000000000

Distributions

- Two different issues issues: Distribution/contributors and users
- Users: External repositories: Browsers addons (100% free isn't sufficient)
- Distribution and users: Nonfree software
- → Less technical users often cannot know how to stay free software
- → Distribution don't want to fix (distribution policies)

Certified distributions

- FSDG: Free system distribution guidelines: https://gnu.org/distros
- Not perfect but can collaborate to fix bugs.
    - Lot of nonfree software removed. Users sometime find some, bugreport and get it removed. Need help from users.
    - Third party repositories: Being worked on, still lot of work to do. Need help from users.

What about "hardware"? Same limitations than OS?

Computer formfactor	Can run free OS?	Other retrictions?
PC with UEFI without restricted boot	Most of them, for now...	Can't change WiFi card, AMT
Huawei smartphone	No: `https://www.xda-developers.com/huawei-stop-providing-bootloader-unlock-codes/`	Cannot change the boot software, other OS loaded
Android	Difficult	usually no root access

Issues: Software everywhere

- Management Engine: another computer to prevent users control:
  https://www.fsf.org/blogs/sysadmin/
  the-management-engine-an-attack-on-computer-users-freedom
- Smartphones: Similar issue: "Hidden" operating system launched by nonfree boot
  software: Example: Mobicore
  https://www.replicant.us/freedom-privacy-security-issues.php
- Raspberry PI: "Small" GPU firmware: full blown operating system is hidden in the
  "small" nonfree GPU firmware: https:
  //ownyourbits.com/2019/02/02/whats-wrong-with-the-raspberry-pi/
- Smartphones: Tracking devices: https://www.gnu.org/proprietary/
  malware-mobiles.html#phone-communications
- Again: not exaustive: [insert your own issue here].

Nonfree boot software + common distro can interact

- Automatic updates with fwupd: Updates sometimes breaking computers, making it run slower, etc.

- Some boot software can also prevent downgrades (The Management engine has that for instance).

How to know it all?

- Computers everywhere even in computers
- No guarantees with common comptuers

Introduction
000

From free software applications down to the hardware
0000000000000000●0

Using the certifications
00000000000

Under the hood and DIY approach
0000000000000000000000000

ARM computers with u-boot
000000000000

Solutions?

- Work to replace nonfree software?
  - High priority free software projects
    https://www.fsf.org/campaigns/priority-projects/
  - Take a long time, not always possible
- Making hardware that works without nonfree software?
  - Not magic: require chips that work with free software already.
  - Can be expensive to get (What if I don't have a lot of money?)
  - Can be difficult to ship (What if I live in Palestine)
  - Can be difficult to procure (States require big companies as small ones can't ship big quantities (like 500000 computers or more) and and companies are to be paid years later.)
- FSF certifications?
  - Works for non technical users (super important) $\rightarrow$ Can scale
  - Help find nonfree software and know the status
  - Also need hardware that works without nonfree software to stay usable.
- $\rightarrow$ We need all of them
  - Know what doesn't work $\rightarrow$ Start to fix
  - Getting help from users (Thinkpenguin, ath9k_htc)

Why hardware and software certifications are complementary?

- Even FSF certified distribution runs nonfree software if the computer is not certified. Examples: ACPI (comes from the BIOS/UEFI, bytecode in the GPUs "BIOS").

- Users with a certified computer and not certified distributions will probably run nonfree software without knowing it at some point, and it's not a bug.

FSF certified hardware and distributions: What is available?

The certifications

- RYF (Respect your freedom): Certification for hardware: `https://ryf.fsf.org`
- FSDG (Free System Distribution Guidelines): Certification for software (distributions): `https://gnu.org/distros`

Certified laptops: https://ryf.fsf.org/index.php/categories/laptops

- Technoetical T400, T400s, T500
- Taurinus, Technoetical, Vikings X200
- Technoetical X200s, X200T

Warning

- Order with an FSDG distribution → FSDG distribution works.

- Third party hardware (graphic card for instance) can still introduce nonfree software.

- Only certifies software freedom, not if the vendor is reliable, cheap, "eco-friendly", does fair-trade, ships on time, etc.

  - → You need to do your own research for the rest (read people's reviews, ask people, make sure the information is up to date, etc).
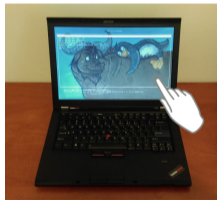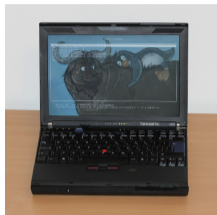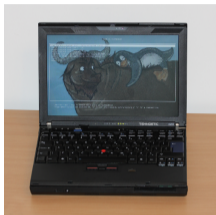
What can you do with these?

- Refurbished computer from circa 2008
    - Enough RAM: Upgradable, up to 8GiB, Buy through RYF vendor or ask to test to make sure it's compatible.
    - CPU still fast enough for most tasks
        - Limits: Videos <1080p, heavy compilations (full distributions from scratch).
    - Mate display and good quality keyboards $\rightarrow$ can do work (writing, programming, etc) on them.
    - Gigabit Ethernet
    - Cheap and/or big storage (2 SATA HDDs/SSDs possible on some laptops).
    - WiFi: 2.4 cheap, 5GHz works in crowded environments (like big conferences).
    - Highly customizable (Some of them can have multiple WiFi cards for instance).
    - Some of only have trackpads, some have both
    - Removable batteries, semi-rugged, last long time. Limits:
        - Ethernet on X200.
        - Requires some light maintenance (removing dust).
- Compatible with all the FSF certified distributions but Replicant

More details: `https://libreplanet.org/wiki/Group:Hardware/Computers/`
`Laptops/Freeable_laptops/Libreboot_Laptops_comparison`

Other type of computers (AKA formfactors)

Certified WiFi access points: https://ryf.fsf.org/index.php/categories/routers

- TPE-R1300 Wireless-N Mini Router
- TPE-R1200 Wireless-N Mini Router v2
- TPE-R1100 Wireless-N Mini Router

What can you do with these?

- Mainly building (home) networks.

- Requires isolation from rain, dust, etc to build community networks.
  https://media.libreplanet.org/u/libreplanet/m/
  freeing-networks-where-we-need-freedom-most/

- Runs LibreCMC:
  - Fully graphical web interface that it easy to use and very robust.
  - Also works with command line or configuration files.
  - Still requires some networking knowledge (what is an IP address? What is "DHCP" ?).

Certified server / workstation: https://ryf.fsf.org/index.php/categories/routers
- Vikings ASUS KCMA D8 Workstation

What can you do with these?

- Also compatible with all the FSF certified distributions but Replicant
    - Way faster than the laptops
    - Best for servers (Display controller limited)

DIY and current status

A look into software projects

- Coreboot
- Libreboot
- U-Boot (and u-boot-libre)
- And FSDG distributions

X86 and reminder

- Experiement: Remove the HDD / SSD, power on your laptop: something shows up on the display (a vendor logo, some text, etc).
- → Software is running, that software is the boot software.
- BIOS, UEFI, Coreboot, Libreboot, etc

Press The ESC Key for Startup Menu
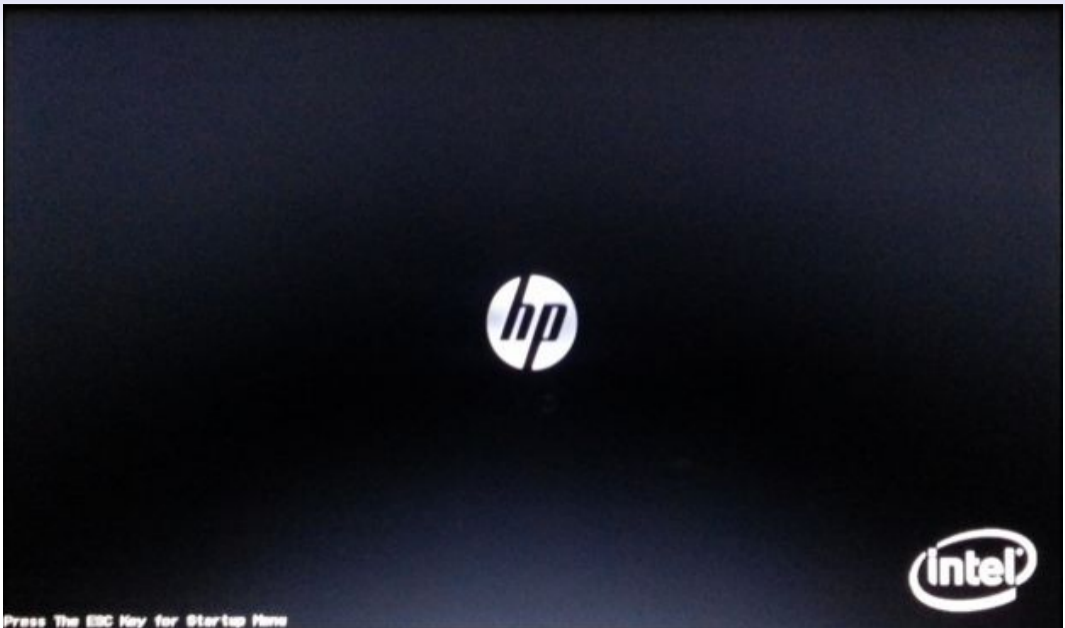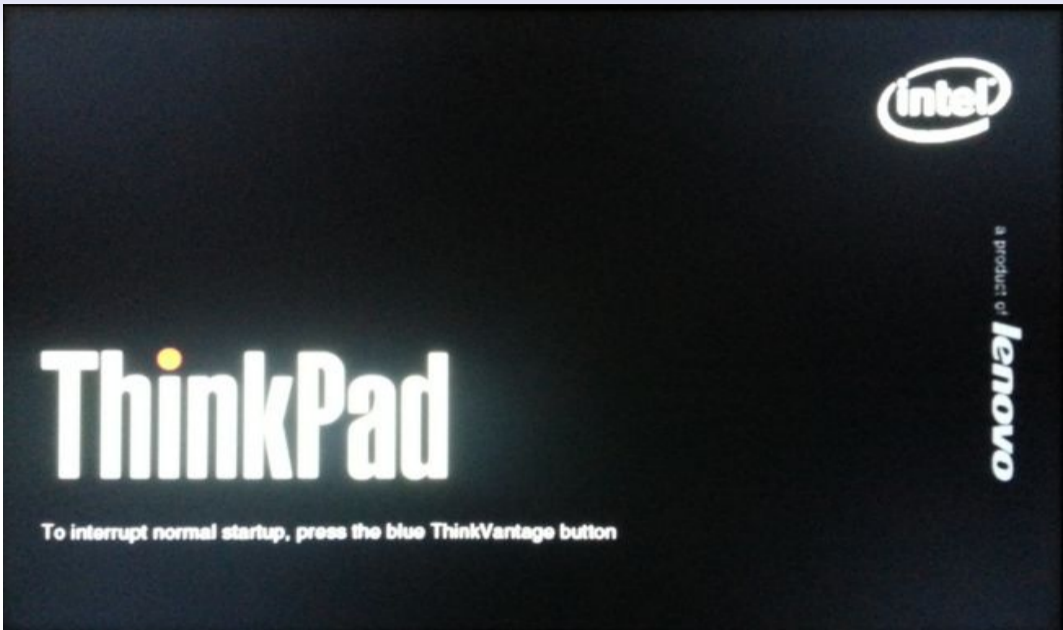
Where is that software?

- In a memory chip, inside the mainboard

Projects

- Coreboot: Almost completely free or completely nonfree software depending on the computer. $\rightarrow$ Not usable as-is.
- $\rightarrow$ Libreboot: Coreboot distribution created in 2009 that solved this issue.
- $\rightarrow$ We'll look into the recent changes in Libreboot later on.

How to use "Libreboot" ?

- Get a supported computer
- Download Libreboot
- For some supported computers: run a script to install it
- For other: disassemble the computer, and reprogram the memory chip inside the mainboard with some dedicated hardware.
- Reassemble the computer, change the battery if needed, Add a WiFi card that works with free software, etc.

ARM and u-boot: what is u-boot

- Boot software
- Replaces the BIOS / UEFI and GRUB
- Can boot many ARM computers without nonfree software.
- U-boot is typically packaged by distributions like Guix, Parabola, PureOS or Trisquel.
- "Libreboot" was interested in adding support for u-boot too.

My story with u-boot ("About me" slide)

- I ended up maintaining Parabola u-boot packages and organizing how to support ARM devices in Parabola with several goals:
    - Lowering maintenance
    - Being able to support a huge quantity of devices
    - $\rightarrow$ Use computers well supported by upstream.
    - $\rightarrow$ Organize the code and documentation to have as little as possible to do for each new device.

Issue: I found nonfree software in u-boot

- Nonfree software found:
  - Licenses/r8a779x_usb3.txt
  - drivers/usb/host/xhci-rcar-r8a779x_usb3_v3.h
  - Debian also found nonfree software in u-boot (drivers/dma/MCD_tasks.c):
    `https://metadata.ftp-master.debian.org/changelogs/main/u/u-boot/`
    `u-boot_2021.01+dfsg-5_copyright`
- Can't redistribute it in Parabola
- Parabola has a mechanism (mksource()) to produce deblobbed tarballs, but
  Parabola packages for u-boot were too complex → didn't work.
- Other FSF certified distributions (like Guix) also use u-boot.
- → Used Libreboot to make a libre u-boot (u-boot-libre) and reused it in Parabola.

PKGBUILD I

```
# Copyright (C) 2019 Denis 'GNUtoo' Carikli <GNUtoo@cyberdimensio
# This program is free software: you can redistribute it and/or m
# it under the terms of the CC0 1.0 License.
# Maintainer: Parabola Hackers <dev@lists.parabola.nu>

pkgname=python-bundlewrap
_pkgname=bundlewrap
pkgver=4.7.0
pkgrel=1
pkgdesc="Agent-less configuration management"
arch=('any')
url='https://bundlewrap.org/'
license=('GPL3')
depends=('python-cryptography'
```

PKGBUILD II

```
            'python-jinja'
            'python-mako'
            'python-passlib'
            'python-pyaml'
            'python-requests'
            'python-six'
            'python-tomlkit')
options=(!emptydirs)
source=("https://files.pythonhosted.org/packages/source/${_pkgnam
sha512sums=('0b12c583bd37ad34e956b4c1c2eed7422acbdef6ed434bf3ce6c

build(){
  cd "$srcdir/bundlewrap-$pkgver"
  python3 setup.py build
```

PKGBUILD III

```
}

package(){
  cd "$srcdir/bundlewrap-$pkgver"
  python3 setup.py install --root="$pkgdir/" --optimize=1 --skip-
}
```

But Libreboot too started adding nonfree software...

- $\rightarrow$ I regrouped together with other people (Like Adrien 'neox' Bourmault) and organization already using computers with Libreboot (Like Libre en communs, Technoetical, others) who needed a free software Boot software.

We are continuing the Libreboot project but witout nonfree software

- Status:
  - Rationale explained at https://libreboot.at
  - Infrastructure: Mailing lists, bug reports, website, git repositories.
  - Status: we accept patches but don't have a release yet.
  - Short term: remake the latest free release.
  - (And also make u-boot-libre releases as well).

Longer term goals: make new releases

- More details at: https://libreplanet.org/wiki/Group:
  Hardware/Upstream_projects/Coreboot

Thinkpads with GM45 chipsets

- Laptops:
  - Thinkpad R400
  - Thinkpad R500
  - Thinkpad T400
  - Thinkpad T400s
  - Thinkpad T500
  - Thinkpad W500
  - Thinkpad X200
  - Thinkpad X200s
  - Thinkpad X200 Tablet
  - ThinkPad X301
- Status:
  - Most patches should be easy to upstream or carry around.
  - 2 patches for not using nonfree microcode updates. 1/2 rebased.

Computers with Intel I945 chipsets

- Laptops:
  - Apple iMac 5,2
  - Apple Macbook 1,1
  - Apple Macbook 2,1
  - Gigabyte GA-G41M-ES2L
  - Intel D945GCLF
  - Thinkpad R60 (with Intel GPU only)
  - Thinkpad T60 (with Intel GPU only)
  - Thinkpad X60
  - Thinkpad X60s
  - Thinkpad X60 Tablet

- Status:
  - Apart from the Gigabyte GA-G41M-ES2L and Intel D945GCLF, no more free software GPU init upstream but the Linux i915 driver seem to be able to initialize the GPU at least for the X60.

Computers with AMD chipsets

- Laptops:
    - KCMA-D8
    - KFSN4-DRE
    - KGPE-D16
- Status:
    - Not supported anymore by upstream coreboot but there is some community support in a coreboot 4.11 branch.

Libreboot compatible ARM computers

- Status at: `https://libreplanet.org/wiki/Group:Hardware/Upstream_projects/Coreboot`
- Laptops:
    - ASUS C201 Chromebook
- Status:
    - Still in Coreboot 4.19 but no maintainers
    - Unknown status in Parabola. (wael in #parabola in liberachat proposed to test).
- Computers with u-boot: we didn't take decisions about them, but probably not the priority as can handled by FSF certified distros as well (with u-boot-libre or some deblob procedure).

Short digression: PowerPC 64bit little endian

- Interesting computers:
    - Raptor Engineering (Talos II, Blackbird, etc), some are RYF: 2 computers in one:
        - BMC (Baseband management controller): small ARM computer to boot the big one: Doesn't depend on nonfree software but no free distro for it.
        - The big powerPC computer might work on Guix and/or on Trisquel 11 (not out yet).
    - Some Single board computers (like the NXP T2080 RDB) or the PowerPC notebook prototypes can run u-boot but display isn't working with free software.

ARM computers with u-boot

- Lot of computers that can boot with free software
- Also lot of them that can't
- No direct competition with Intel, limitations (RAM).
- Lot of hardware variation (microSD, SATA, displays, no displays, etc).
- $\rightarrow$ Need consideration when buying (use cases, microSD less reliable, etc).

Hardware constraints

- Boot medias:
  - SD / MicroSD
  - Internal memory (eMMC)
  - "Boot flash"
  - SATA (quite rare)
  - "Boot order"

Hardware and software constraints

- Bootrom / Boot ROM → First stage bootloader → Second stage bootloader → Linux + small filesystem in RAM (initramfs / initrd) → init (GNU/Linux).

- Issue: First stage bootloader at different offsets

- Issue: A single U-boot image cannot work yet on many computers.

- → Most of the time the distribution has bootloader packages but only a generic installer that doesn't boot on microSD at best.

Distributions

- Guix (32bit currently broken, 64bit seems to work)
- Parabola (32bit works, 64bit computers support outdated, need help)
- PureOS: 64bit only
- Trisquel 10: 32bit
- Trisquel 11: 32bit, 64bit

Parabola: How to add an ARM computer

- Add a package for a bootloader: (requires to know the computer name, and the u-boot configuration, if it's similar to one already supported).
- Document the computer in the wiki:
  - Add information about how to install Parabola on it
  - Describe it a bit (amount of RAM, name, etc)
  - Describe what works what doesn't and the required hardware (HDMI display, keyboard or serial console cable).
  - Add that computer to the list of officially supported ARM computers

Parabola: Officially supported

- https://wiki.parabola.nu/Computers
- Olimex: Lime 1 A20: Full support
- Olimex: Lime 2 A20: Full support, 1 bug to fix
- Beaglebone black: Partial support (no 3D acceleration, video decoding, PRUs, capes, etc).
- → Many more just need documentation / installation instructions

Parabola: Software status

- No support for accessories (capes, shields etc) yet

- Only support extlinux.conf and old style booting (u-boot boot protocol)

- No UEFI + GRUB support yet

Guix: Software status

- ARM 32bit currently broken
- Diffucult to customize (kernel modules) but good abstraction (image types for specific computers).
- Also produces images for the Pinebook PRO but only in "latest" (no signatures), though easy to build them yourself.

PureOS and Trisquel

- Very similar: Both are directly or indirectly derived from Debian.
- Similar situation: No documentation → Where to add it? The Libreplanet wiki?
- Need user testing and fixing
- Different policies: can contribute to Trisquel directly, PureOS prefers people to contribute directly to debian whenever possible (lowers maintenance).
- Some features make it harder to support: Multiple kernel but no way to update /boot/extlinux/extlinux.conf → Use UEFI?

Tests with Trisquel

- Ideally have a bootloader image in PureOS or Trisquel (in one of u-boot-sunxi, u-boot-img, u-boot-omap, etc)
- "Device tree" shipped by the kernel
- Using 1 kernel to support updates
- Writting some configuration to make it boot and update well
- $\rightarrow$ Tried rapidely with a Lime 1 A20: could't find the devicetree.
- $\rightarrow$ Tried rapidely with a Beagleboard: could't find u-boot.img.
- $\rightarrow$ Tried rapidely with a Pandaboard: hang during boot (no rootfs found?) lot of stacktraces.
- $\rightarrow$ I also have a TBS2910: no bootloader
- $\rightarrow$ We should probably retry with Trisquel 11

Call for action and advertizements

- Try to have more collaboration between FSDG distributions:
  - https://libreplanet.org/wiki/Group:Hardware
  - https://libreplanet.org/wiki/Group:Software
- Help for testing ARM, improving documentation, etc.
- Help Parabola or FSDG distributions that are understaffed

Licenses of this presentation

Other Pictures