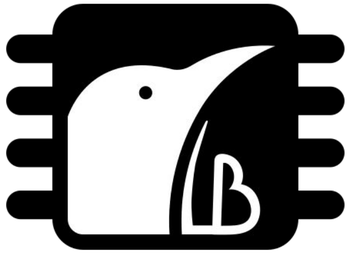
Open. Together.

OCP
SUMMIT

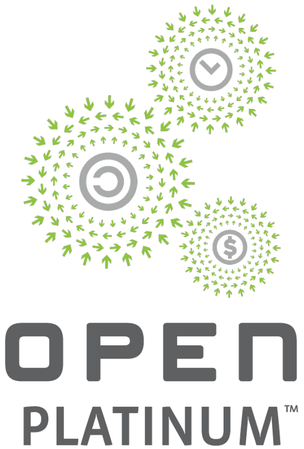


OSF/Security

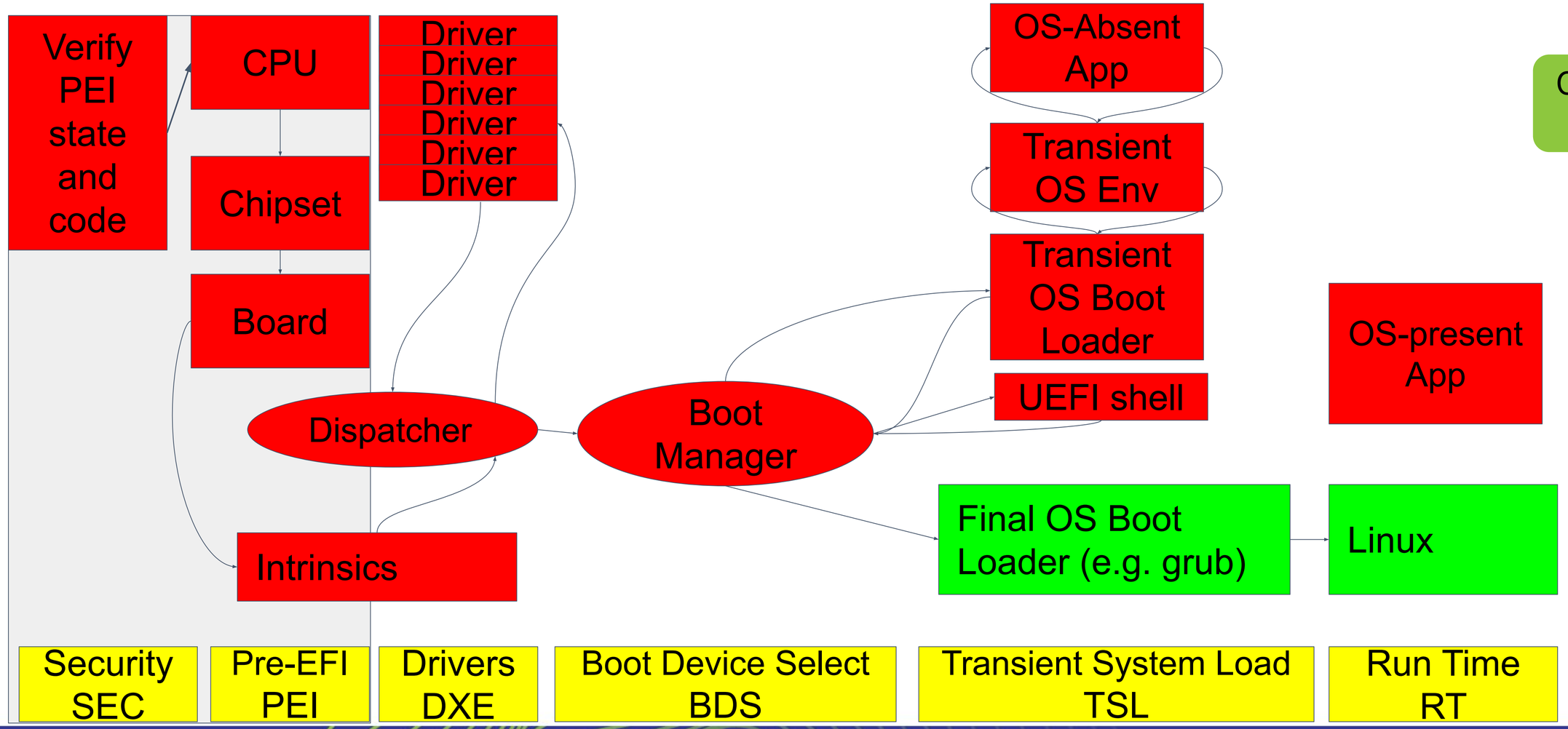
LinuxBoot: Boot anything from Linux

Chris Koch (twitter.com/hugelgupf) — Google
with Max Shegay — Google

with Ron Minnich, Ryan O'Leary, Gan Shun Lim — Google
with Trammell Hudson — Two Sigma
with Jean-Marie Verdun, Guillaume Giamarchi — ITrenew
with David Hendricks, Andrea Barberio, Tobias Fleig, Łukasz Siudut — Facebook
with Philipp Deppenwiese — 9elements Cyber Security
and many others



UEFI: Closed Source in red



OPEN SYSTEMS FIRMWARE

Embedded Software

UEFI Ecosystem

OPEN SYSTEMS FIRMWARE

- UEFI Implementations
  - are mostly closed source,
  - written in C,
  - share an address space in ring 0.

- Vendors are incentivized to ship it and forget.
- **Owners do not own their system.**

Embedded Software

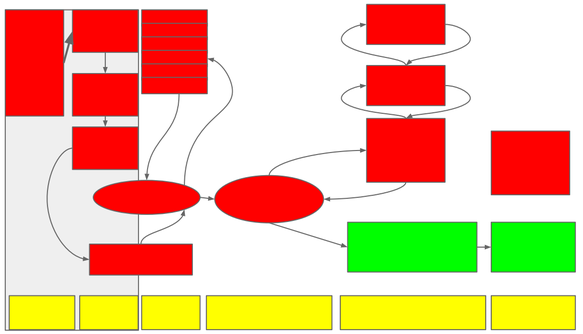
UEFI Ecosystem

OPEN SYSTEMS FIRMWARE

- 20+ vendors involved in shipping firmware
  - IBVs (BIOS vendors),
  - silicon manufacturers,
  - ODMs, OEMs,
  - NIC, disk, BMC, … vendors,
  - OS vendors (Windows, RHEL, Debian)
- Black boxes that wrap black boxes.
- **What happens with vulnerabilities?**
  - Who owns fixing it?
  - How to integrate it?
  - Goes through 3-5+ layers of vendors. Release?

Embedded Software

LinuxBoot on UEFI



Verify PEI state and code

CPU

Chipset

Board

Intrinsics

Dispatcher

Driver

Driver

Driver

U-root (Go programs) Netboot or local boot

Boot Kernel Linux

OS

Security SEC	Pre-EFI PEI	Drivers DXE	Boot Device Select BDS	Transient System Load TSL	Run Time RT

LinuxBoot

OPEN SYSTEMS FIRMWARE

- Linux has problems, too!
  - Yes, but it's **open, measurable, reproducible, updatable.**
  - Has drivers for everything.
  - Has 1000s of contributors constantly working on vulnerabilities, improvements, ...
- Kernel Engineers = Firmware Engineers.
  - How many of your SREs or Sysadmins know Linux vs UEFI?
- Speed
  - Winterfell boot time: **8 minutes to 20 seconds.**
  - DXE dispatcher is trial and error: no runtime dependency tree.

Embedded Software

Linux as a Bootloader

OPEN SYSTEMS FIRMWARE

- Use Linux in firmware to boot the OS
- Trivial to use modern features
  - HTTP(S), IPv6, GPG, …
- No more Option ROMs!
- Boot Linux from Linux?
- Boot ??? from Linux?

Embedded Software

Writing a Linux Bootloader

OPEN SYSTEMS FIRMWARE

- sys_kexec_load
  - Specify which kernel pieces are to be moved where on execution.
    - Give a list of segments.
      - logical address {to, from}
      - physical address {to, from}
  - Specify entry point.

Embedded Software

Go Library to the Rescue

OPEN SYSTEMS FIRMWARE

- Developed Tools to deal with kexec
  - Segment allocation, deduplication, etc
  - See https://godoc.org/github.com/u-root/u-root/pkg/kexec
  - See https://godoc.org/github.com/u-root/u-root/pkg/multiboot

- Should make it easier to write LinuxBoot bootloaders for new OS!
- Should make it easier to boot Linux on other archs
  - ARM support in the works

Embedded Software

Multiboot Kernels

OPEN SYSTEMS FIRMWARE

- Max spent a few months working on multiboot kernel support
- QEMU + GDB = Lots of fun!
- Can now boot
  - Akaros
  - Harvey
  - tboot
  - VMware ESXi

Embedded Software

Demo

OPEN SYSTEMS FIRMWARE

Embedded Software

Future Work: Windows

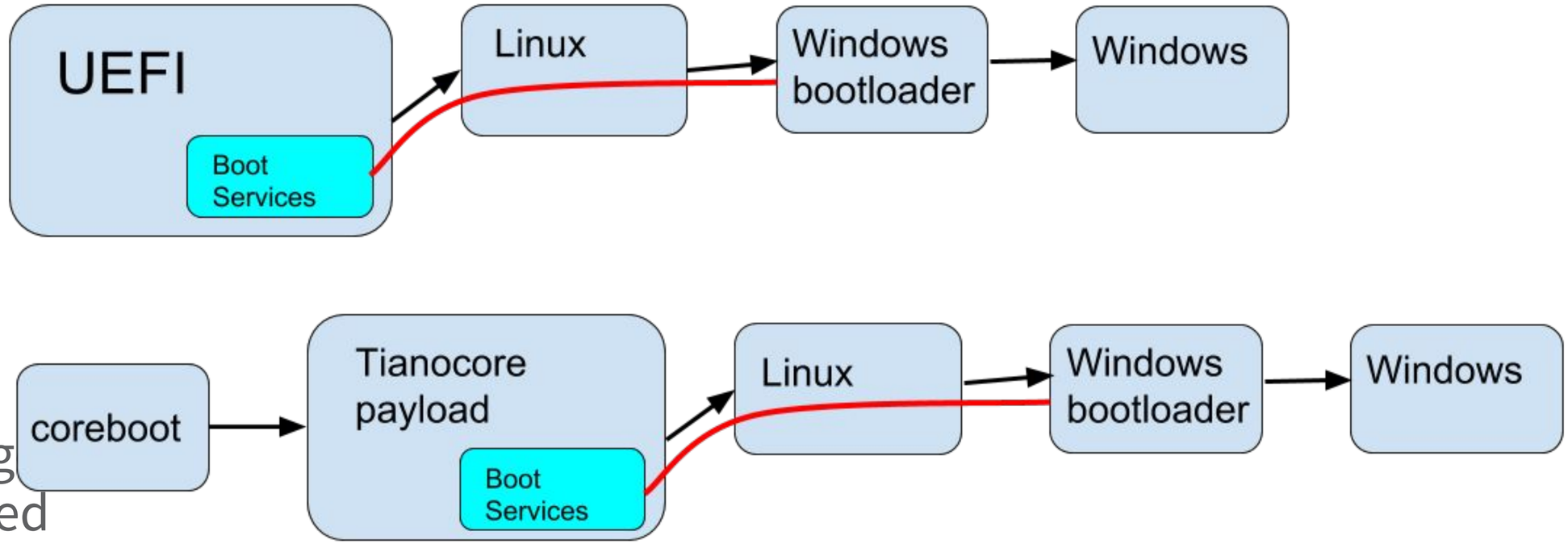
OPEN SYSTEMS FIRMWARE

- Some crazy ideas…
- EFI apps need RuntimeServices
  - Stubs that talk to BMC?
- EFI apps need BootTimeServices
  - Windows wants access to
    - Graphics
    - Disk
    - Network
    - …
    - (On server, which ones do you *actually* need?)

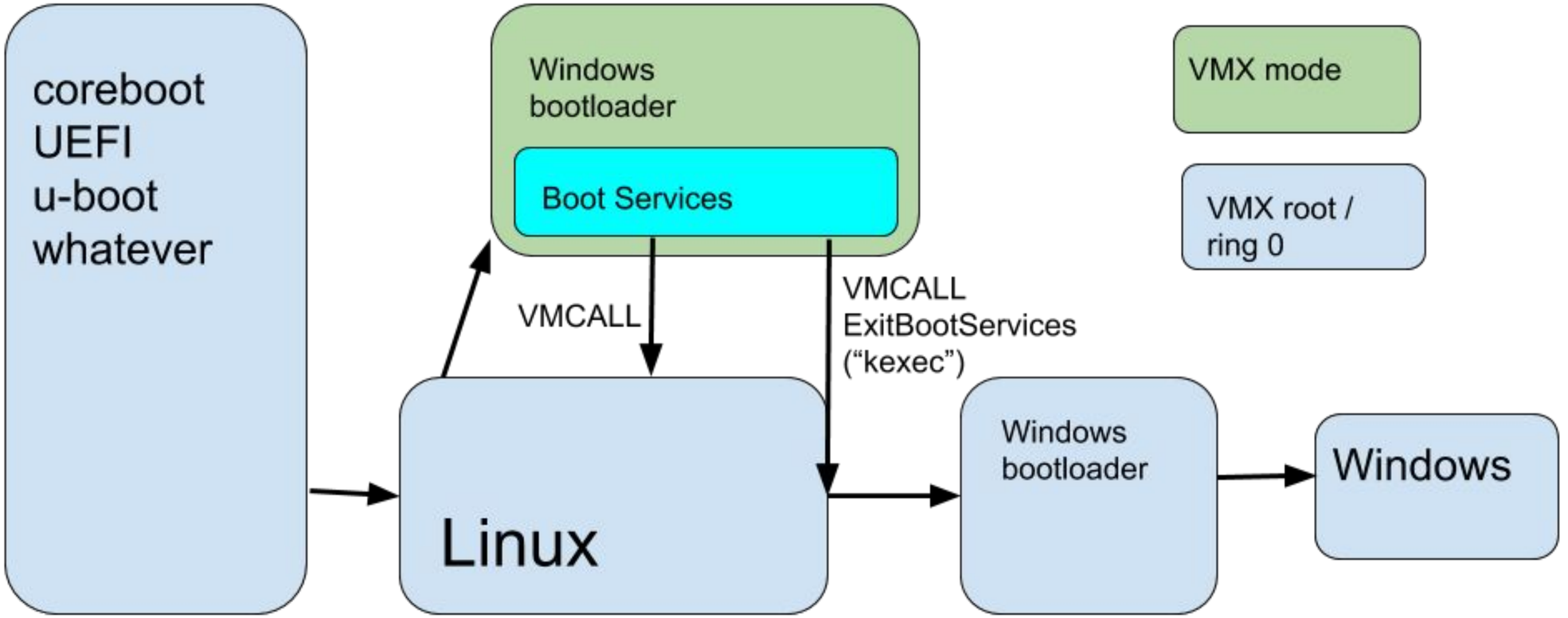
Embedded Software

Booting Windows, Option 1

- **Linux as EFI chainloader**
- Have to keep EFI drivers :(
- Does not fit the “Let Linux do it” model
- coreboot ramstage can’t be eliminated



Booting Windows, Option 2



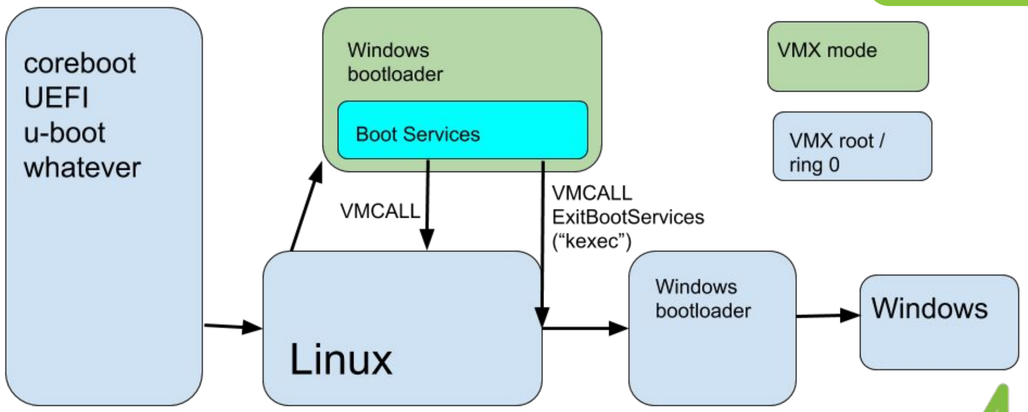
OPEN SYSTEMS FIRMWARE

Embedded Software

Booting Windows: Option 2

OPEN SYSTEMS FIRMWARE

- Start Windows in Hypervisor, emulate everything like Host
  - CPUID just like host, …
- EFI stub implementations in Rust
  - Make syscalls to host using vmcalls
  - Read from disk? VMCALL SYS_READ



- "Dune": Small Linux Kernel Hypervisor to dispatch syscalls
- On ExitBootServices, "kexec" the VM into ring 0
  - Yes, it's nuts.

Embedded Software

Project Guide

OPEN SYSTEMS FIRMWARE

- **Tools, Not Policy.**
  - Foster a community that develops tools.
  - You pick and choose which ones you want in which configuration.

- **Security *and* User Freedom.**
  - Orthogonal to LinuxBoot: security features should allow change of ownership; **reprovisioning hardware** with your own keys.

- Have tools for: **Boots, Not Bricks.**
  - Scary Screen?

Embedded Software

Call to Action

Join Open Source Firmware Slack
https://u-root.slack.com

Join using https://slack.u-root.com

LinuxBoot
https://www.linuxboot.org
https://github.com/linuxboot/linuxboot

u-root
https://github.com/u-root/u-root

LinuxBoot Book
https://github.com/linuxboot/book
See the UTK chapter.

New Hardware
We'll help get LinuxBoot working on your hardware.

Laptop Stickers


Open. Together.
OCP Global Summit | March 14–15, 2019
OCP
SUMMIT