ETH zürich

Systems@ETH zürich DINFK

seL4 as a CPU Driver

Roman Meier, Zikai Liu, Ben Fiedler, Timothy Roscoe

Supported by and in collaboration with Google

Takeaways

The Problem:

1. seL4 hardware assumptions don't match reality
2. Modern SoCs are distributed systems of untrustworthy firmware

Our Approach:

1. Formalize hardware and derive explicit, formal software trust relationships
2. Run seL4 as a secure CPU Driver with assumptions derived from actual hardware

Roman Meier

Zikai Liu

We are looking for your Input! And collaboration!

Come talk to us!

Takeaways

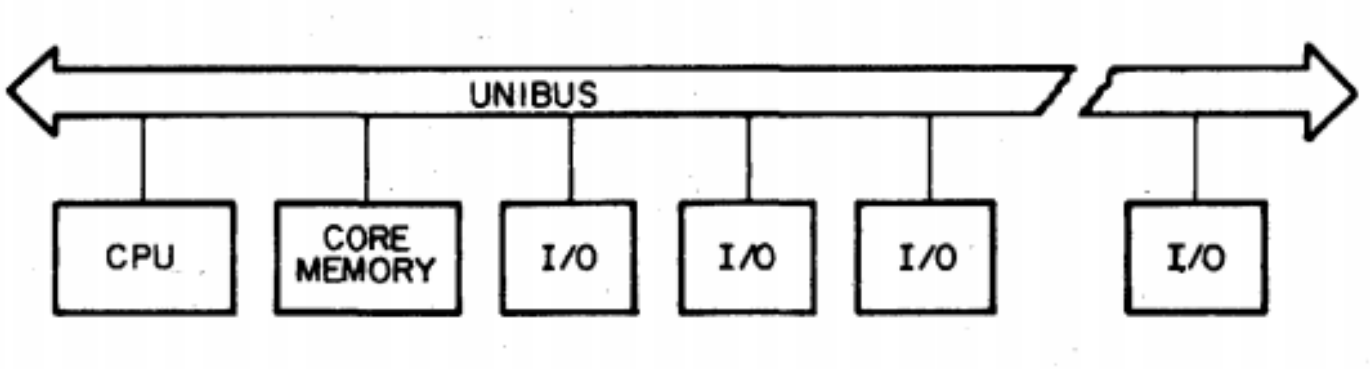
The Problem:

1. seL4 hardware assumptions don't match reality
2. Modern SoCs are distributed systems of untrustworthy firmware

Our Approach:

1. Formalize hardware and derive explicit, formal software trust relationships
2. Run seL4 as a secure CPU Driver with assumptions derived from actual hardware

The modern Hardware View



The seL4 DMA assumptions in detail

“[…] DMA devices are either not present or do not misbehave,
for instance by overwriting the kernel.”
[https://sel4.systems/Info/FAQ/proof.html, 08.10.2024]

Real Hardware is Weirder!

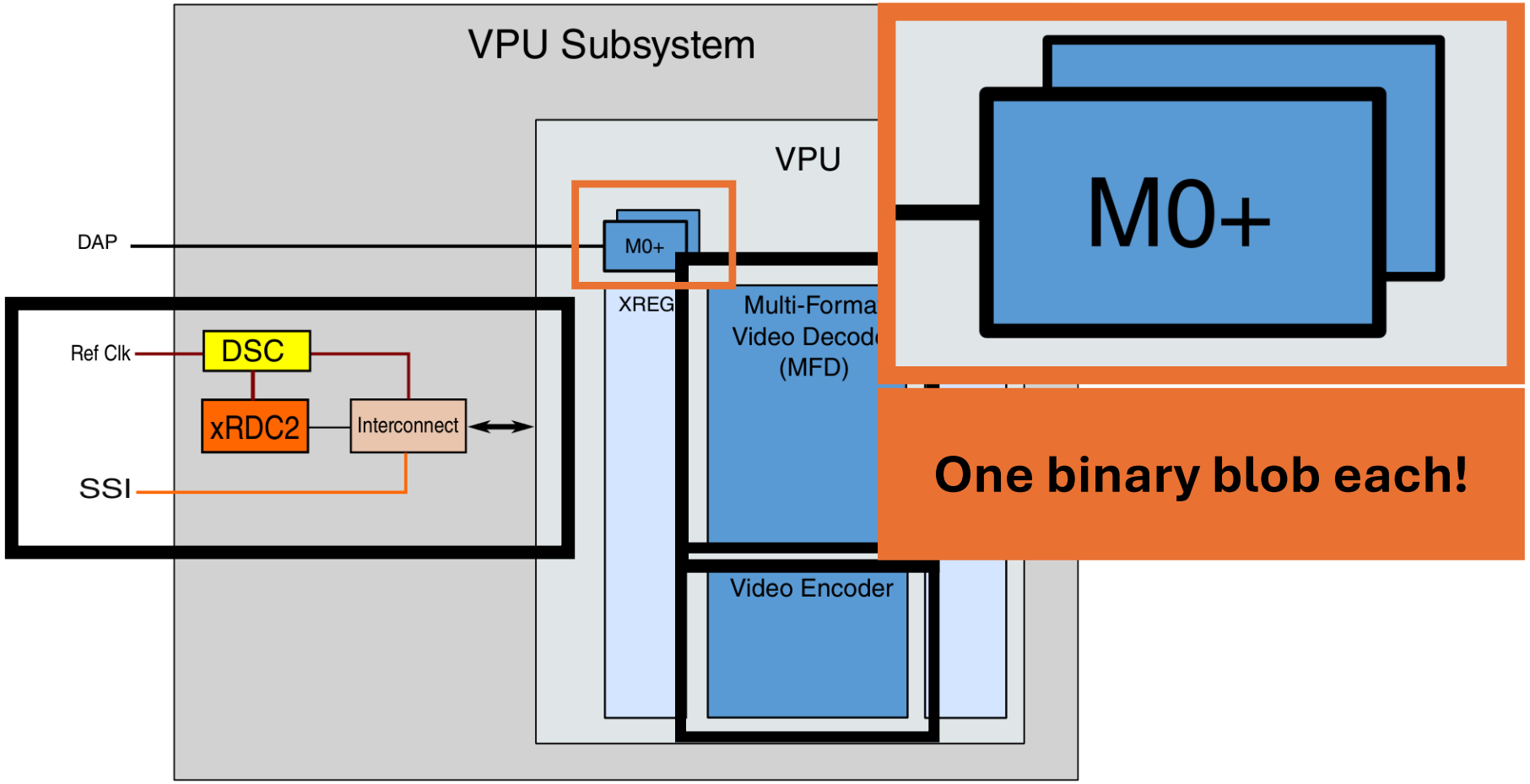


Figure 17-2. Simplified Block Diagram

IMX8DQXPRM, Rev.0, 05/2020

Real Hardware is Weirder!

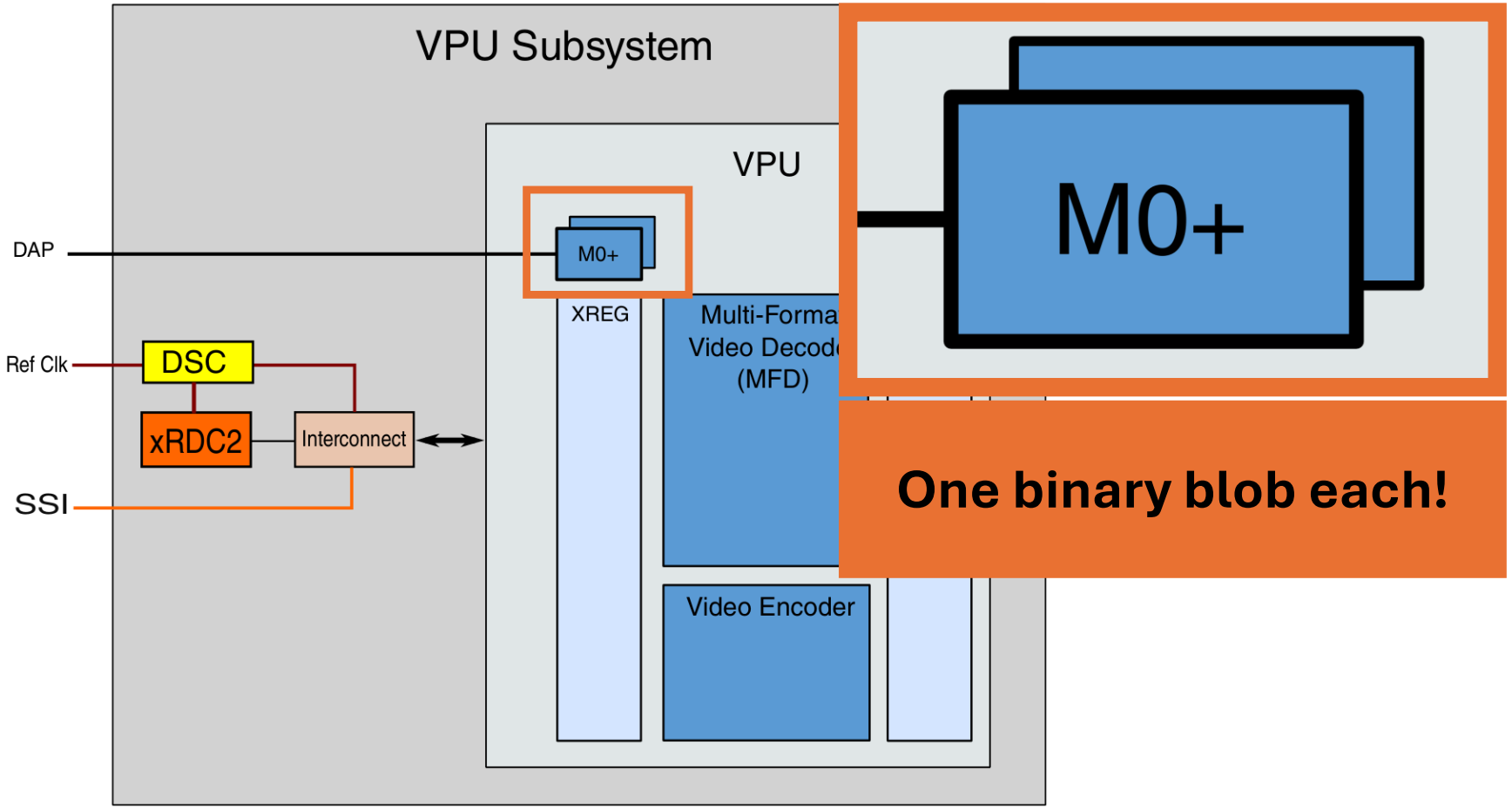
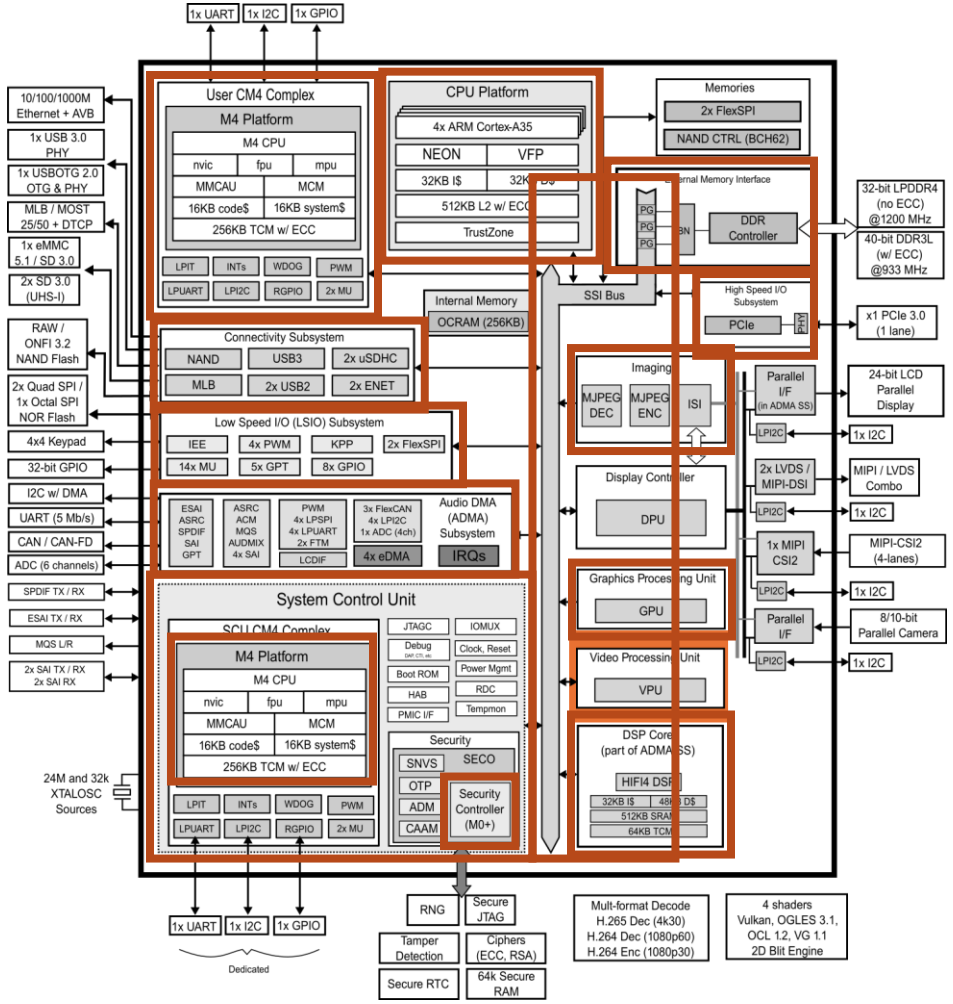


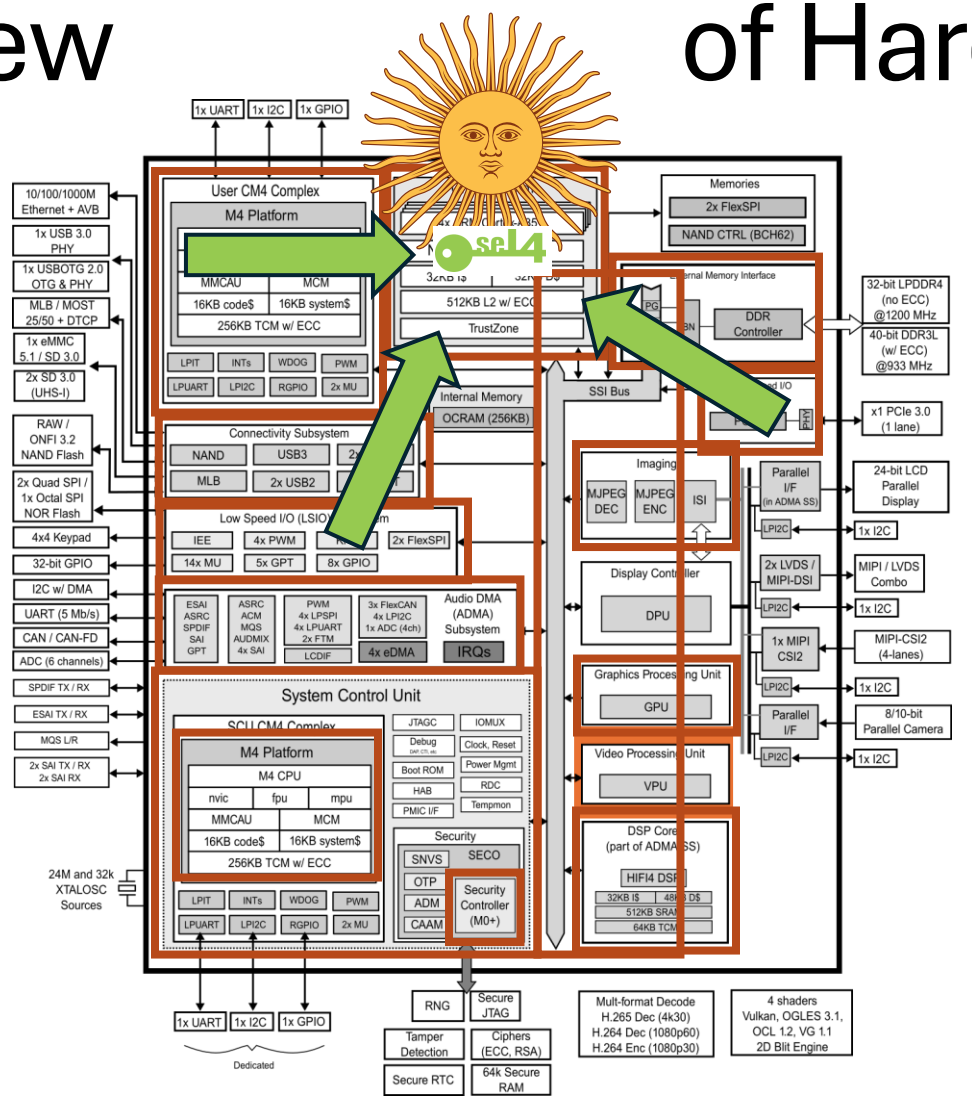
Figure 17-2. Simplified Block Diagram

IMX8DQXPRM, Rev.0, 05/2020

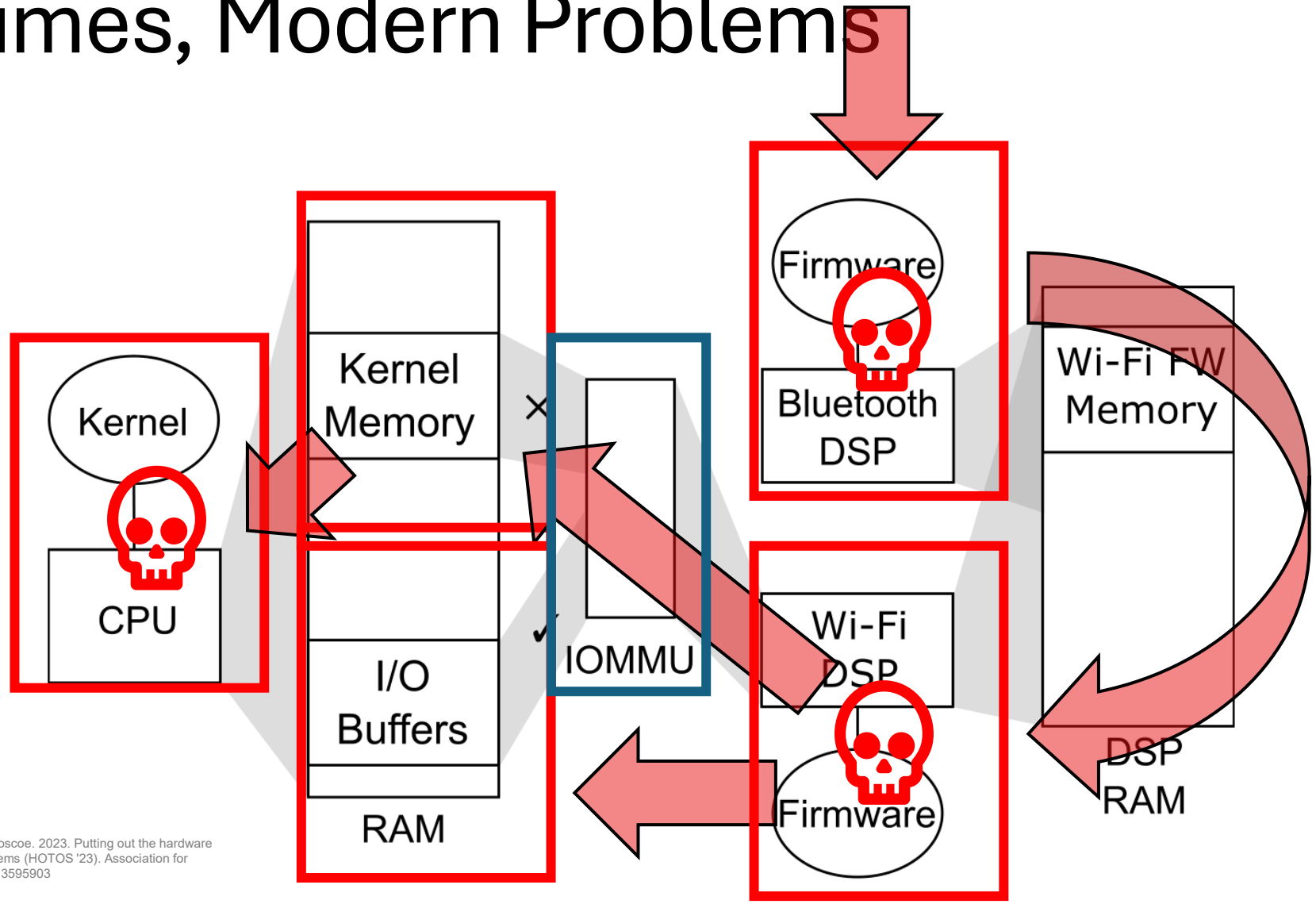
A modern View of Hardware



A modern View of Hardware

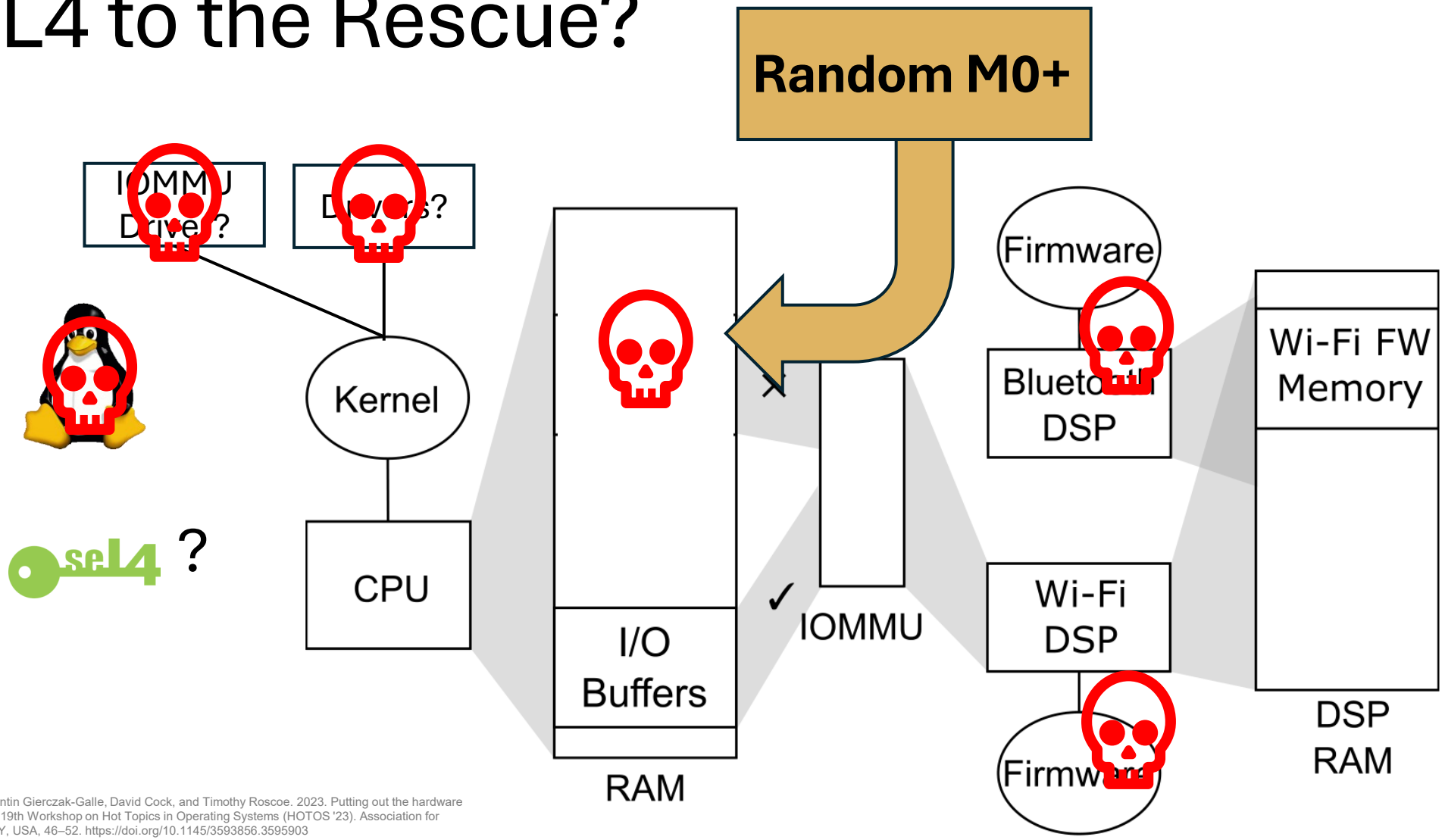


Modern Times, Modern Problems



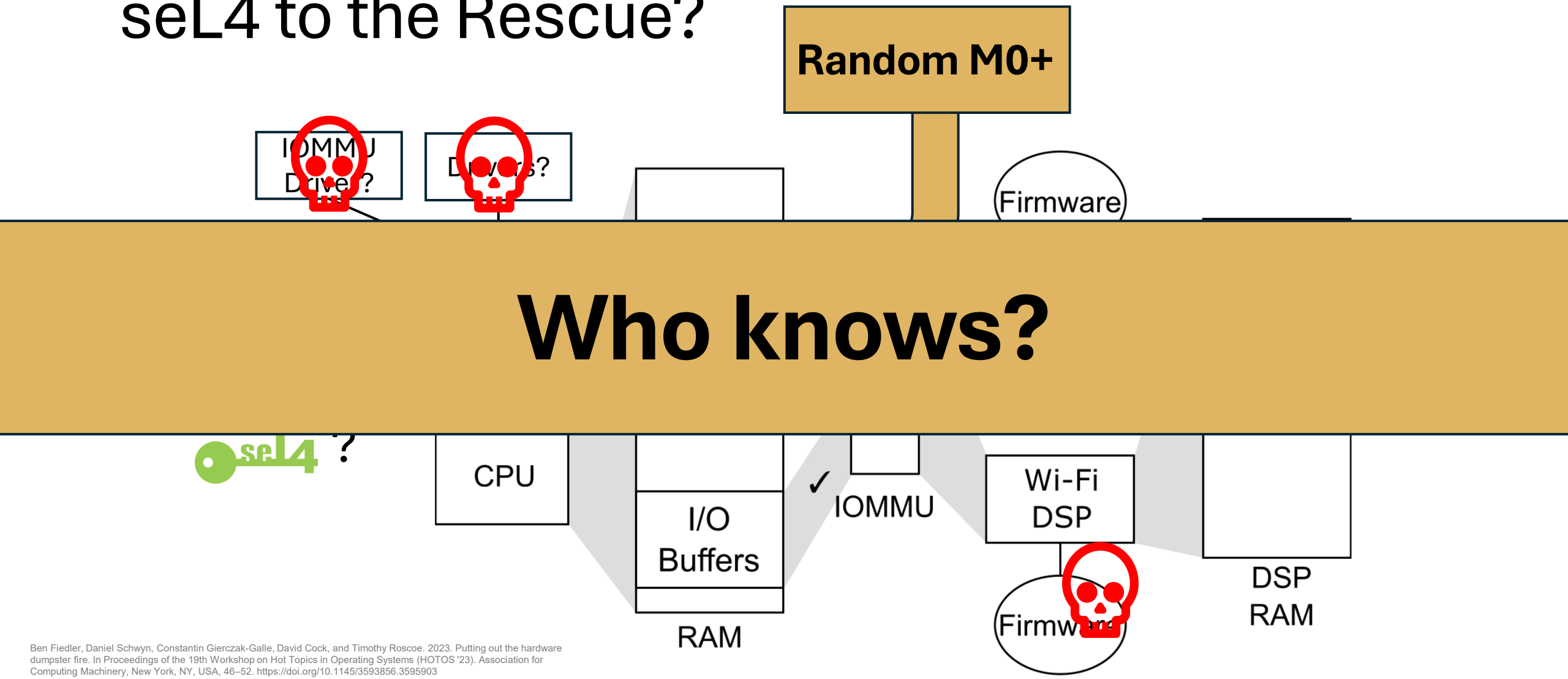
Ben Fiedler, Daniel Schwyn, Constantin Gierczak-Galle, David Cock, and Timothy Roscoe. 2023. Putting out the hardware dumpster fire. In Proceedings of the 19th Workshop on Hot Topics in Operating Systems (HOTOS '23). Association for Computing Machinery, New York, NY, USA, 46–52. https://doi.org/10.1145/3593856.3595903

seL4 to the Rescue?



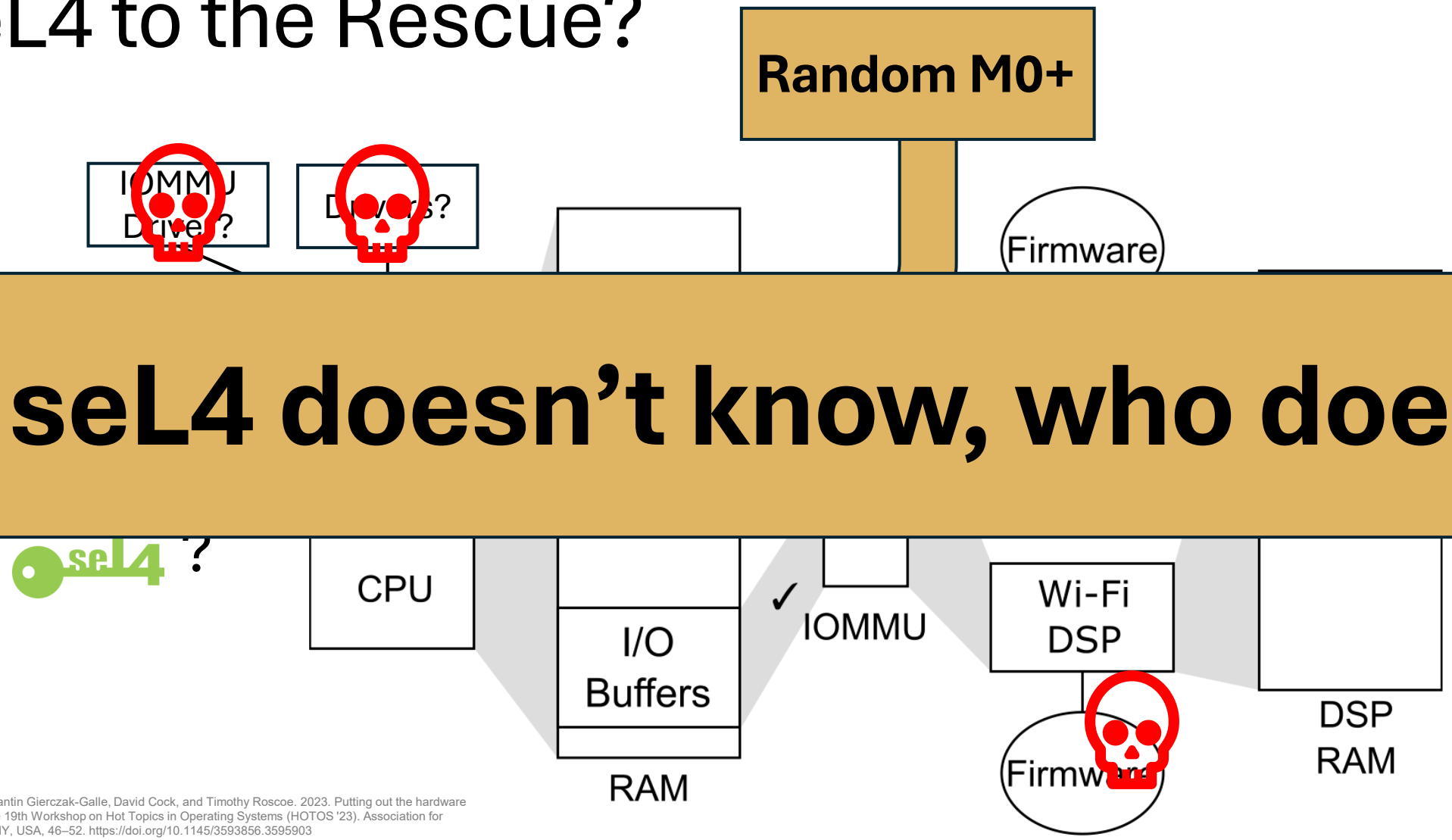
Ben Fiedler, Daniel Schwyn, Constantin Gierczak-Galle, David Cock, and Timothy Roscoe. 2023. Putting out the hardware dumpster fire. In Proceedings of the 19th Workshop on Hot Topics in Operating Systems (HOTOS '23). Association for Computing Machinery, New York, NY, USA, 46–52. https://doi.org/10.1145/3593856.3595903

seL4 to the Rescue?



Ben Fiedler, Daniel Schwyn, Constantin Gierczak-Galle, David Cock, and Timothy Roscoe. 2023. Putting out the hardware dumpster fire. In Proceedings of the 19th Workshop on Hot Topics in Operating Systems (HOTOS '23). Association for Computing Machinery, New York, NY, USA, 46–52. https://doi.org/10.1145/3593856.3595903

seL4 to the Rescue?



If seL4 doesn't know, who does?

Ben Fiedler, Daniel Schwyn, Constantin Gierczak-Galle, David Cock, and Timothy Roscoe. 2023. Putting out the hardware dumpster fire. In Proceedings of the 19th Workshop on Hot Topics in Operating Systems (HOTOS '23). Association for Computing Machinery, New York, NY, USA, 46–52. https://doi.org/10.1145/3593856.3595903

The seL4 Proofs are powerful, but…

- seL4 proofs are very powerful

- seL4 proofs make many assumptions explicit

Modern Operating Systems have a blind spot for modern hardware

Takeaways

The Problem:

1. seL4 hardware assumptions don't match reality
2. Modern SoCs are distributed systems of untrustworthy firmware

Our Approach:

1. Formalize hardware and derive explicit, formal software trust relationships
2. Run seL4 as a secure CPU Driver with assumptions derived from actual hardware

Our Approach

Takeaways

The Problem:

1. seL4 hardware assumptions don't match reality
2. Modern SoCs are distributed systems of untrustworthy firmware

Our Approach:

1. Formalize hardware and derive explicit, formal software trust relationships
2. Run seL4 as a secure CPU Driver with assumptions derived from actual hardware

Kirsch: A real OS with a formal Hw Model

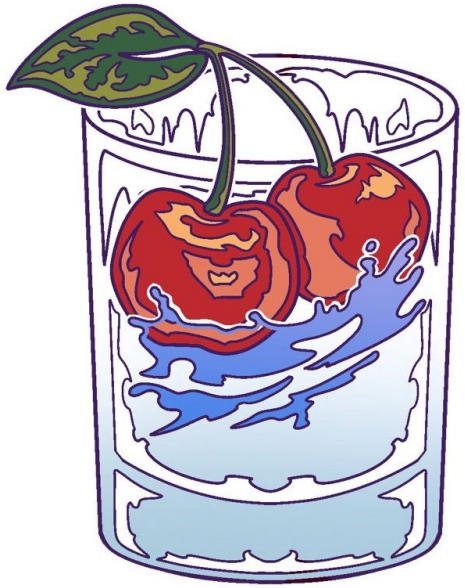
- End Goal: Replace de-facto OS

- Support legacy proprietary Firmware
- **Adapt kernels like seL4**
- Write custom kernel(s)

- Understand and manage complete SoC
- Based on **formal hardware models**

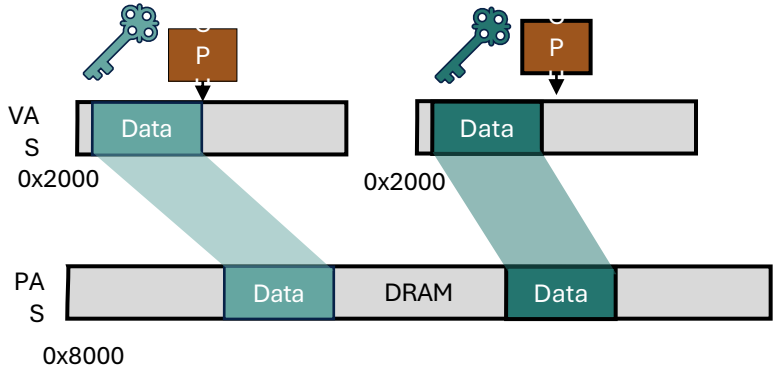
High-level Method

- Explicit, Formal software trust relationships based on actual hardware
- Write down hardware spec
- Derive that each inter-software access either
  - Is impossible
  - Possible, and implies trust

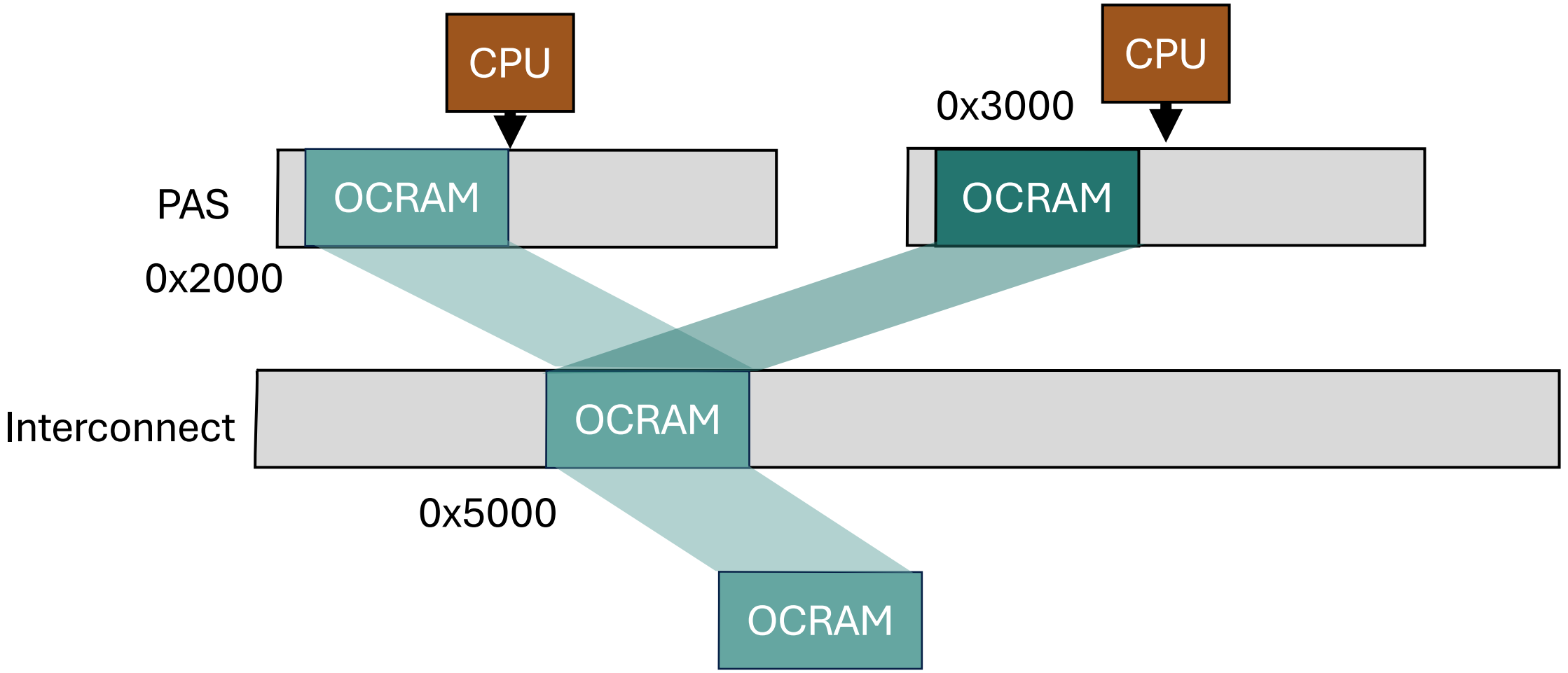


We write a formal Spec for the Hardware

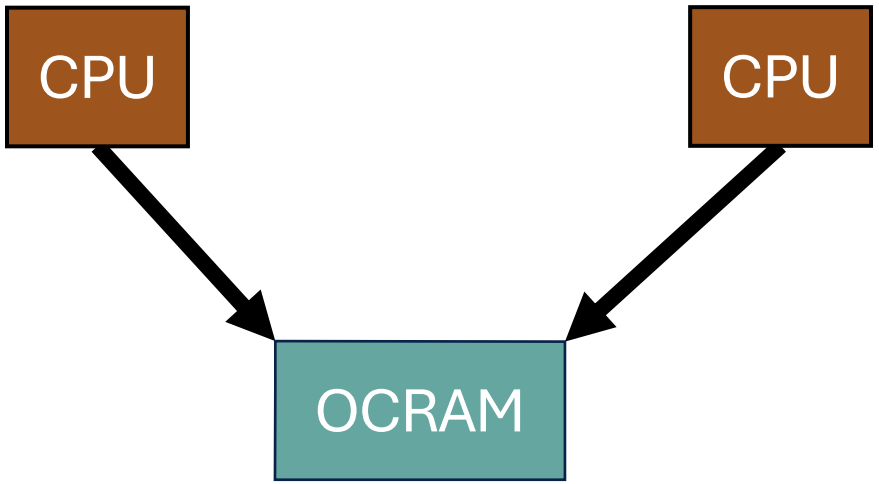
- Write down hardware addressing and translation structure
  - Based on manufacturer documentation
  - Worst case assumptions



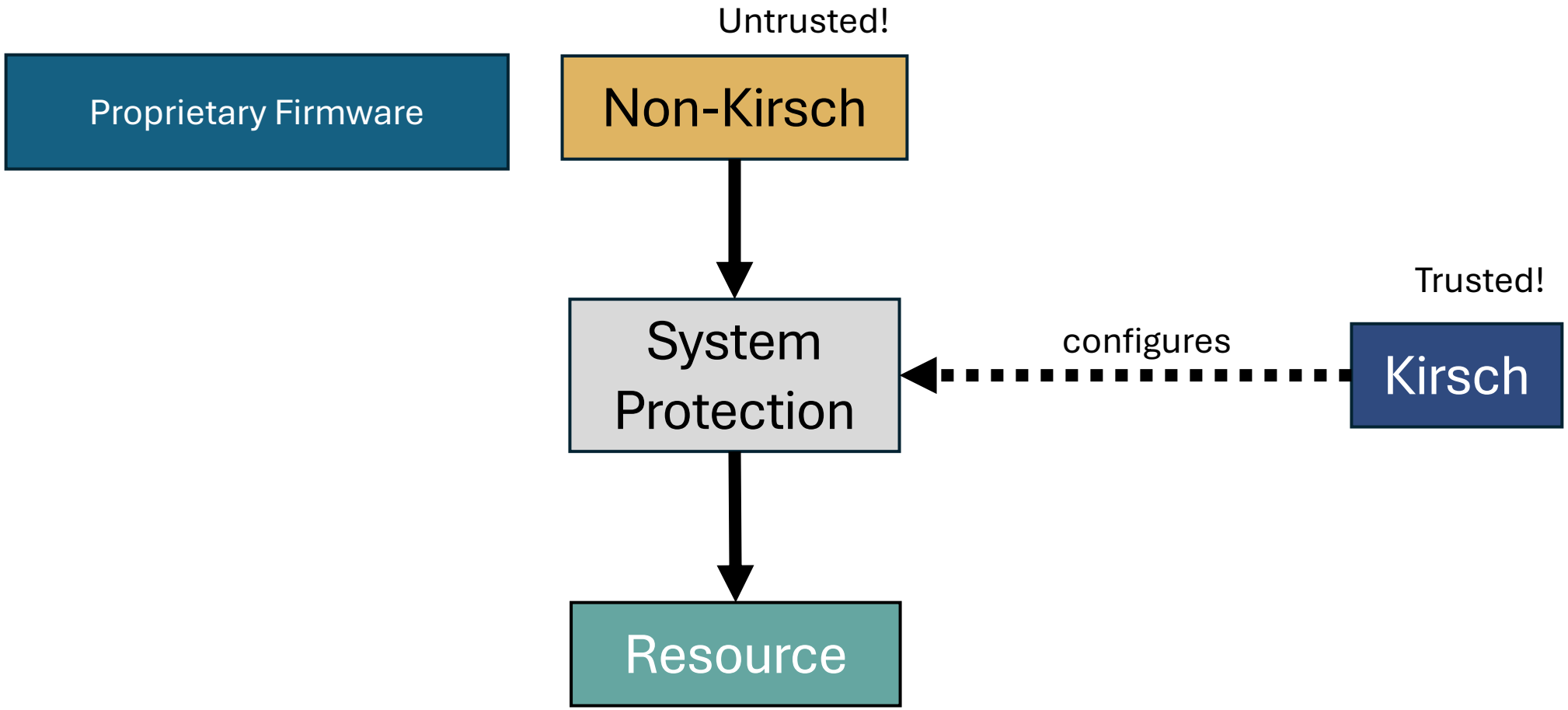
We write a formal Spec for the Hardware



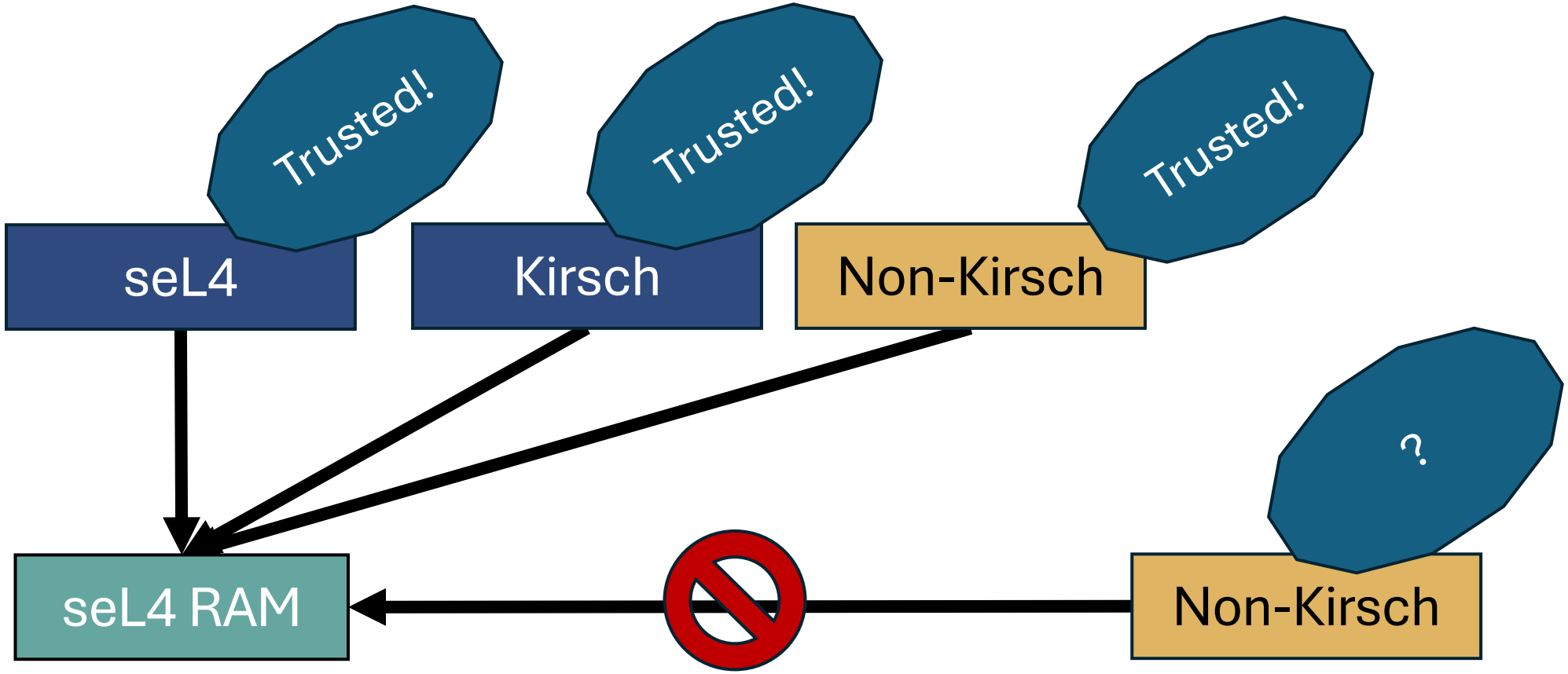
We write a formal Spec for the Hardware



Non-Kirsch Cores



seL4 as a CPU Driver



Takeaways

The Problem:

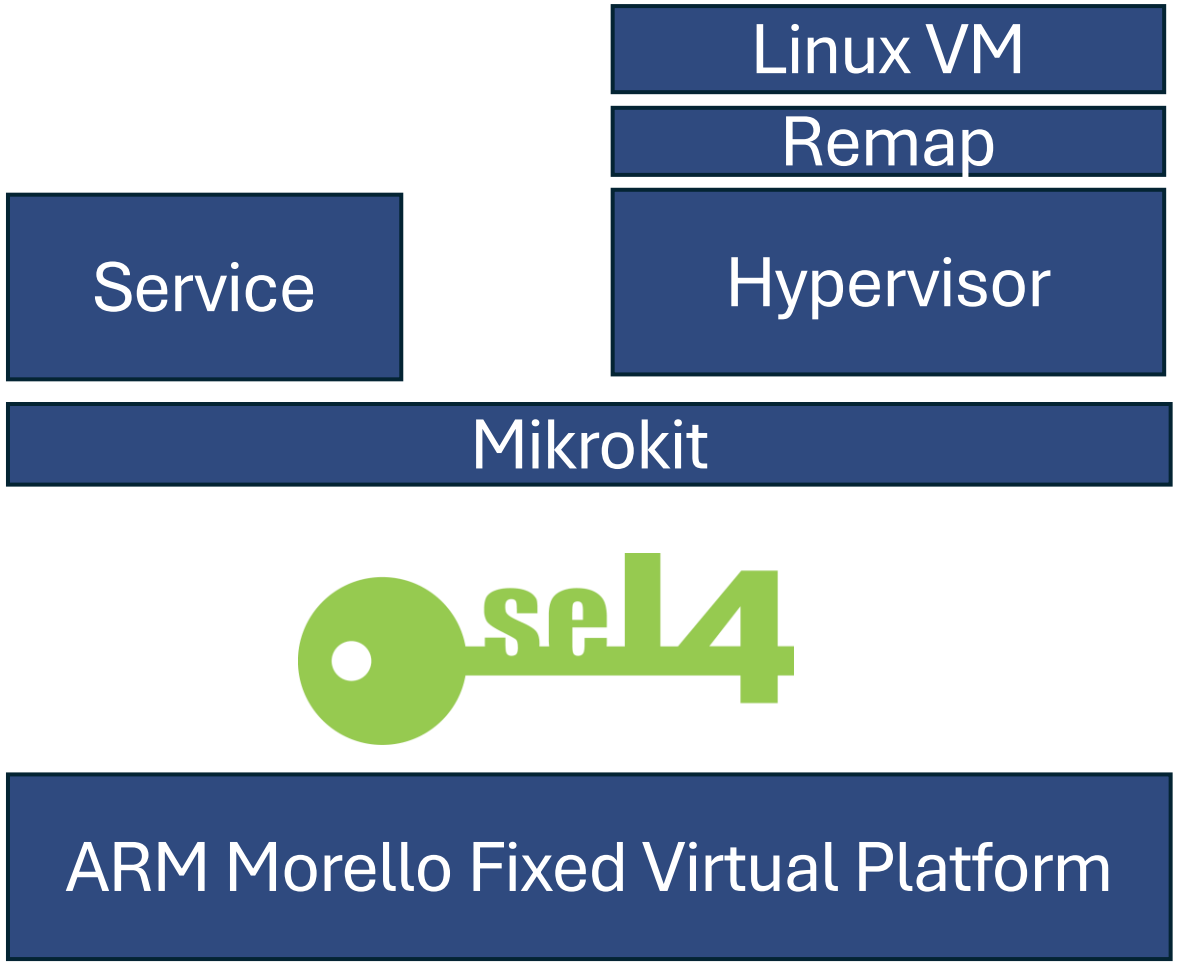
1. seL4 hardware assumptions don't match reality
2. Modern SoCs are distributed systems of untrustworthy firmware

Our Approach:

1. Formalize hardware and derive explicit, formal software trust relationships
2. Run seL4 as a secure CPU Driver with assumptions derived from actual hardware

Current Status

seL4 as a CPU Driver



Sockeye3: Formal HDL

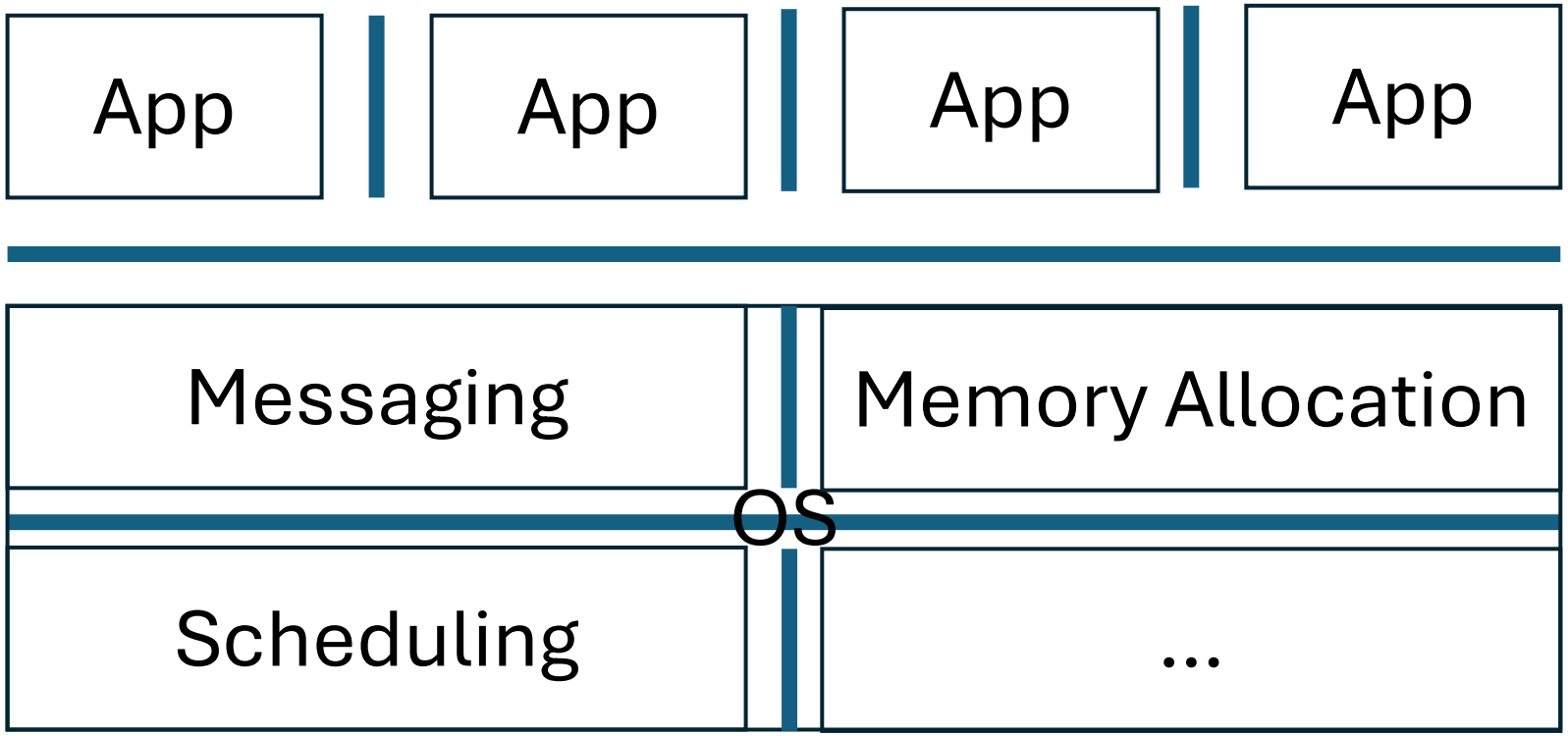
- Custom modular spec DSL
- Based on Decoding Nets
- Interactive and Generative
- SMT Solver Backend
- Under development!

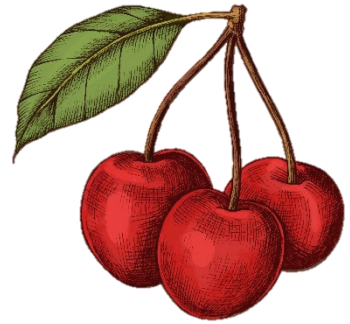
```
res vpu_dec 0x8_0000
res vpu_enc 0x20_0000
res vpu 0x400_0000

map input @ 0x180000 -> vpu_dec @ 0x0 (0x80000)
map input @ 0x800000 -> vpu_enc @ 0x0 (0x200000)
map input @ 0x2c000000 -> vpu @ 0x0 (0x400_0000)

out vpu_dec
out vpu_enc
out vpu
```

CHERI





Future Work

- Formalizing more Platforms
- Extending Sockeye3 with finer reasoning
  - Execution Levels, page rights, etc.
  - (IO)MMU modelling
- Trust Enforcement through SMMU/IOMMU configurations
- CHERI OS with Kirsch integration

Takeaways

The Problem:

1. seL4 hardware assumptions don't match reality
2. Modern SoCs are distributed systems of untrustworthy firmware

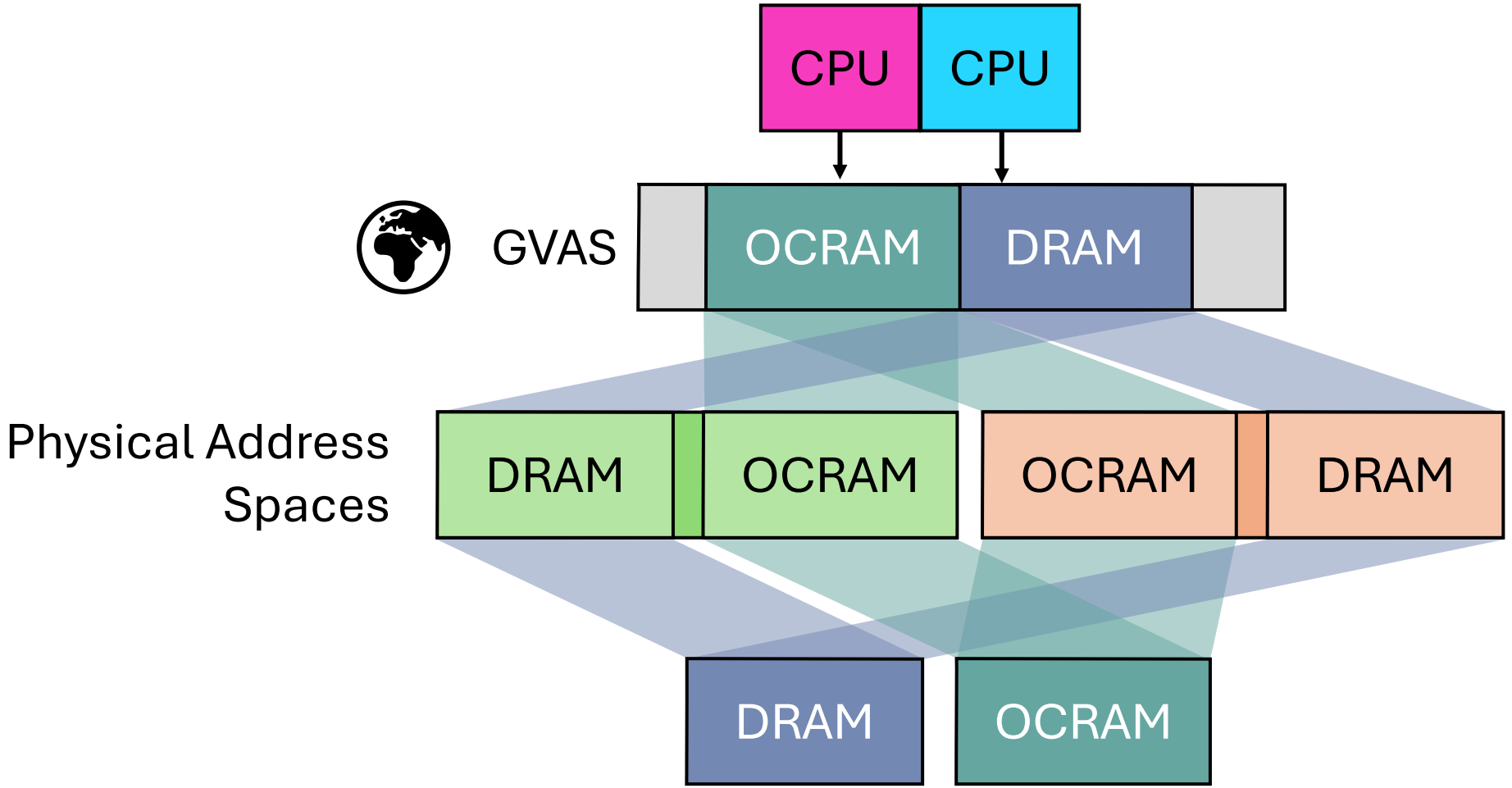
Our Approach:

1. Formalize hardware and derive explicit, formal software trust relationships
2. Run seL4 as a secure CPU Driver with assumptions derived from actual hardware

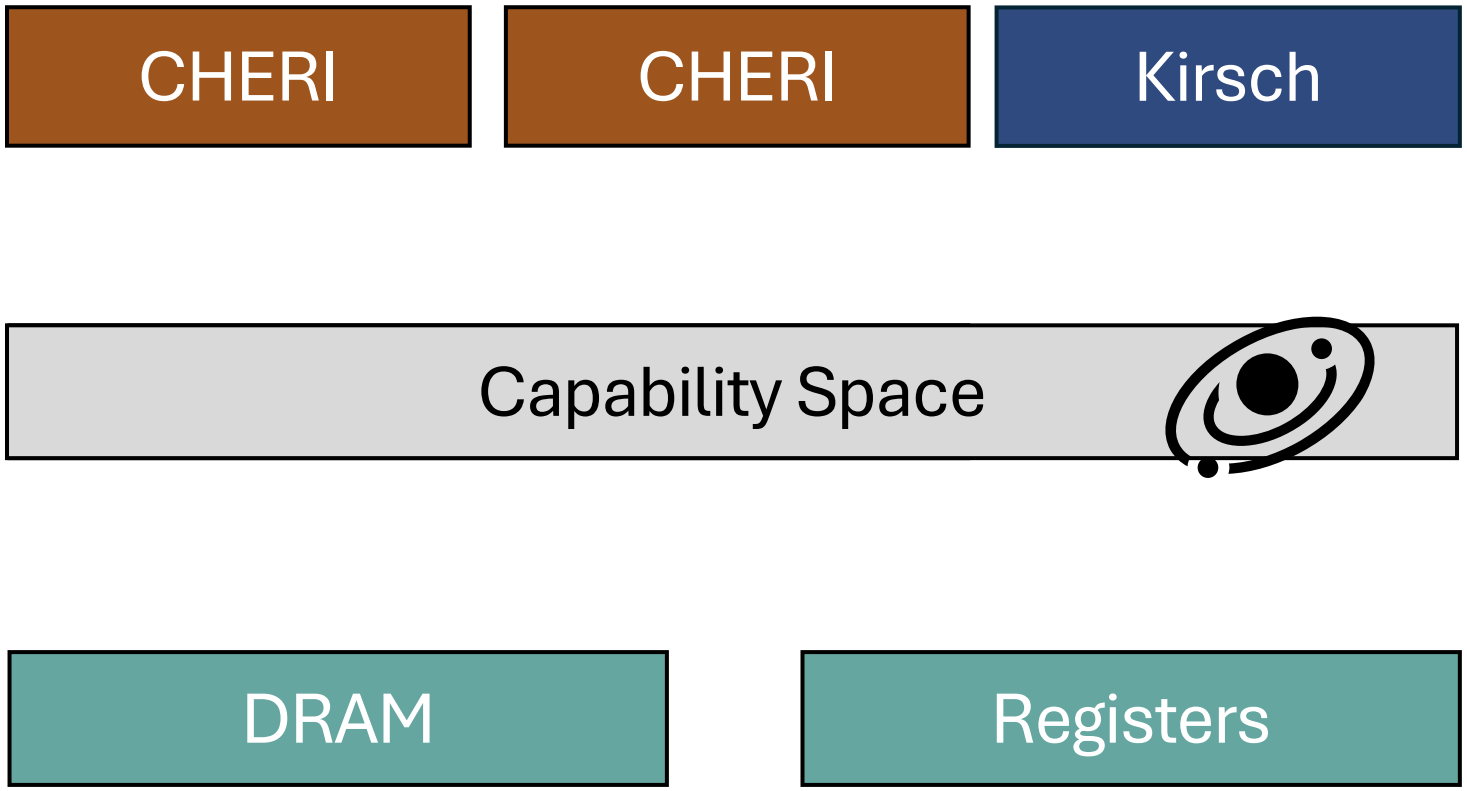
Project Source: https://gitlab.inf.ethz.ch/project-opensockeye

Backup Slides

Global Logical Address Space



Shared Capability Space



i.MX 8X: Page Eight Thousand and Eighty Four

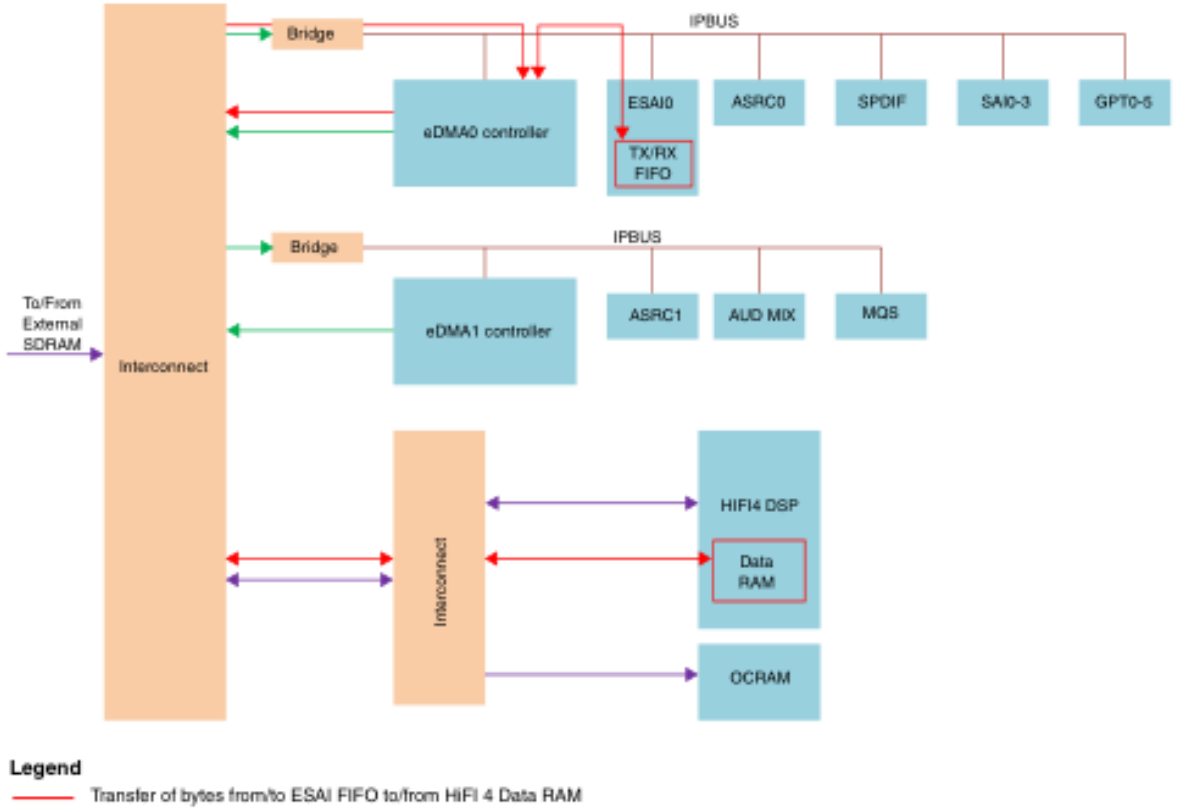
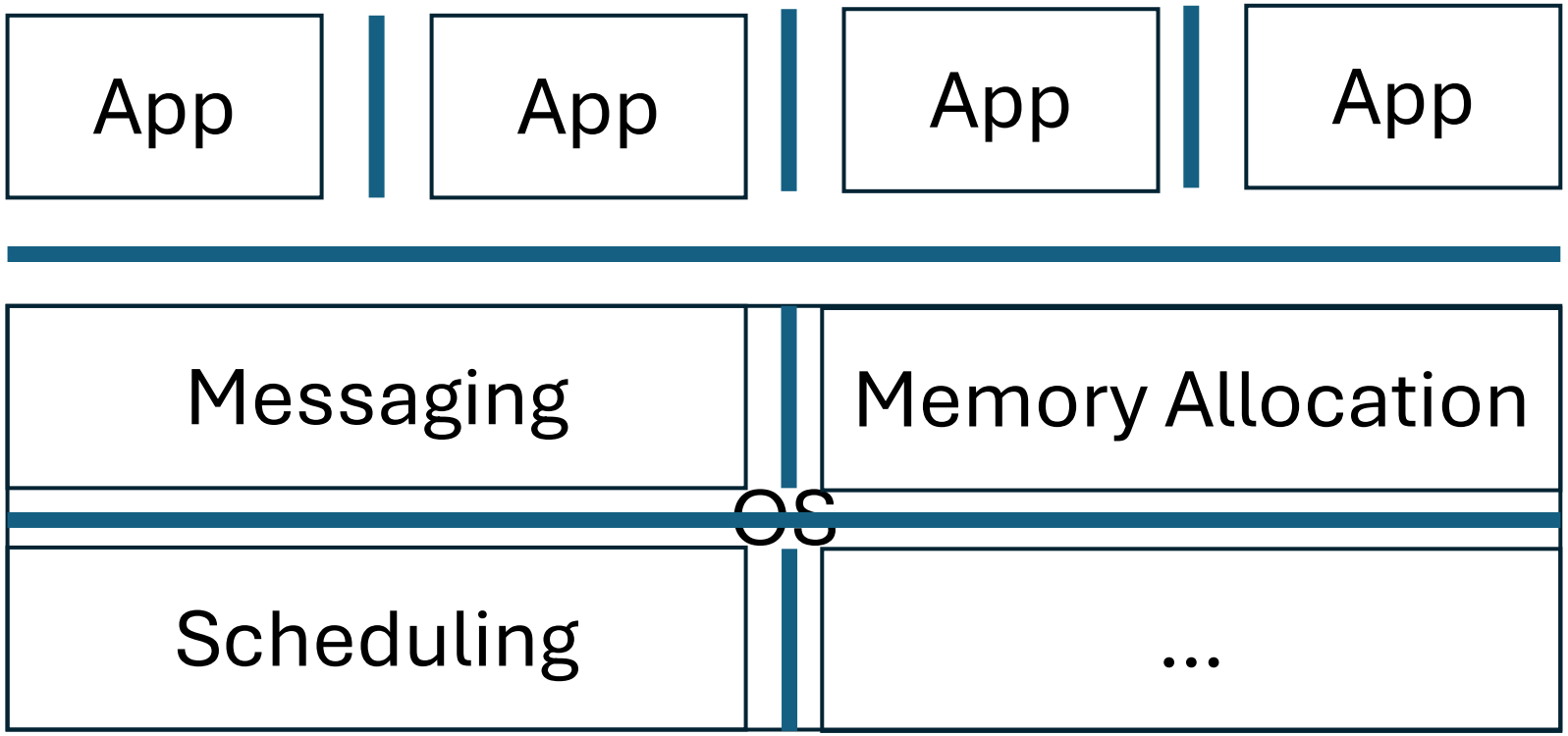
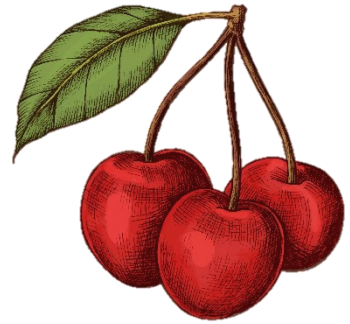


Figure 16-203. ESAI audio routing to HiFi 4

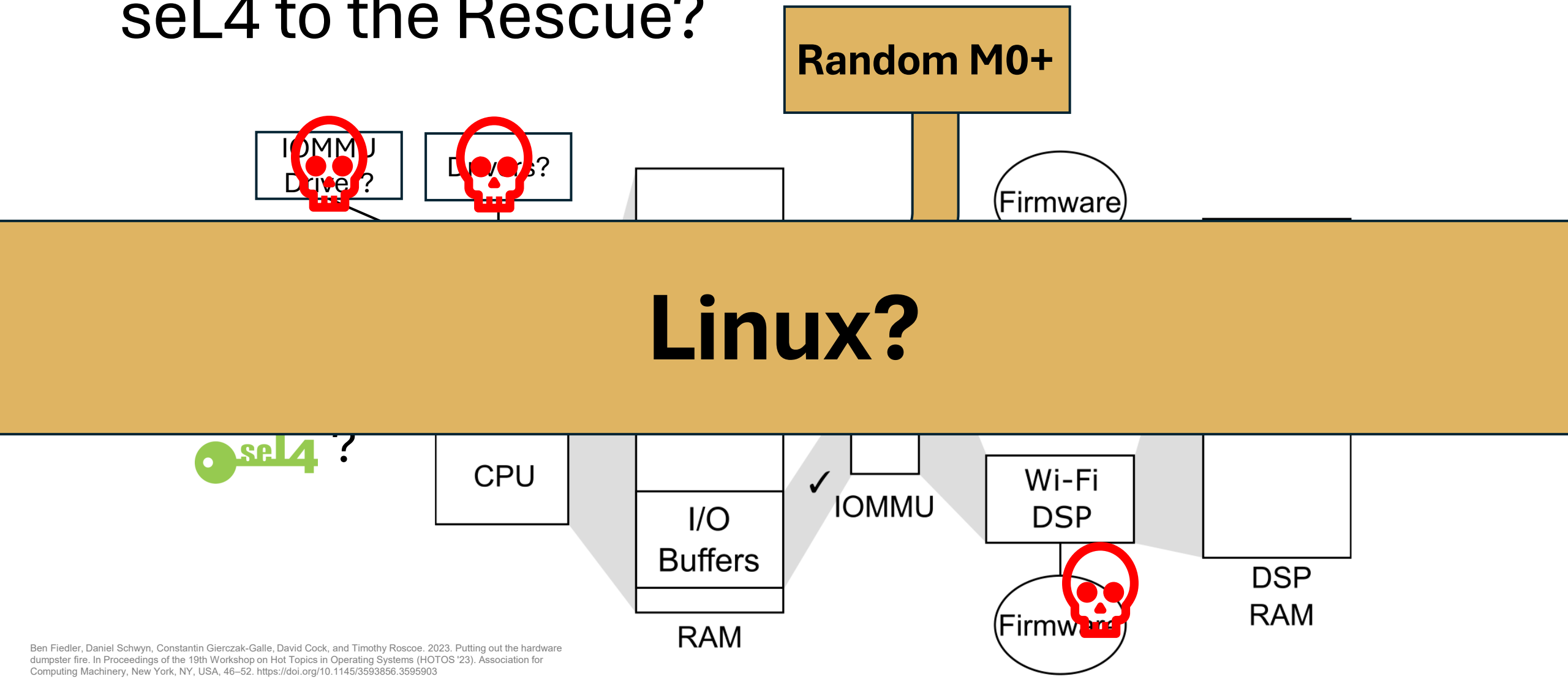
p. 8084

CHERI





seL4 to the Rescue?



Ben Fiedler, Daniel Schwyn, Constantin Gierczak-Galle, David Cock, and Timothy Roscoe. 2023. Putting out the hardware dumpster fire. In Proceedings of the 19th Workshop on Hot Topics in Operating Systems (HOTOS '23). Association for Computing Machinery, New York, NY, USA, 46–52. https://doi.org/10.1145/3593856.3595903