seL4: Verified Operating System for the Real World

Gernot Heiser
gernot@unsw.edu.au
@GernotHeiser

Born August 2009

A NICTA bejelentette a világ első, formális módszerekkel igazolt,

Slashdot

Stories Recent Popular

Betriebssystem mit Korrektheitsb

Forscher am Australia's ICT Research Centre of Excellence (NICTA) haben einen Betriebssystem-Microkernel entwickelt, dessen Korrektheit sie formal beweisen konnten. Der Kernel namens Secure Embedded L4 – kurz seL4 – besteht aus 8700 Zeilen C-Code und soll sich für reale (Embedded-)Anwendungen wie Fahr- oder Flugzeugsteuerungen eignen. Der Beweis umfasst jedoch nur 7500 Zeilen, der Rest ist Boot-Code, der nur einmal ausgeführt wird. Laut dem zwölfköpfigen Forscherteam rund um

Dr. Klein wurd
vergleichbare
wiesen. Es ha
um einen der
sierten Beweis
Beweis behar
Einzeltheorem
mehr als 200
gentliche Bew
ein Programm
übernommen
München und
Cambridge en
Der formale
dass der C-C
exakt der Sy

New Scientist
Saturday 29/8/2009
Page: 21
Section: General News
Region: National
Type: Magazines Science / Technology
Size: 196.31 sq.cms.
Published: -----S-

The ultimate way to keep your computer safe from harm

FLAWS in the code, or "kernel", that sits at the heart of modern computers leave them prone to occasional malfunction and vulnerable to attack by worms and viruses. So the development of a secure general-purpose microkernel could pave the

just mathematics, and you can reason about them mathematically," says Klein.

His team formulated a model with more than 200,000 logical steps which allowed them to prove that the program would always behave as its

Does it run Linux? "We're pleased to say that it does. Presently, we have a para-virtualized ver

Dossiers

code

ogy a világon elsőként b
es ellenőrzését.

lási körülmények közé
goldásokba -, ahol a bea

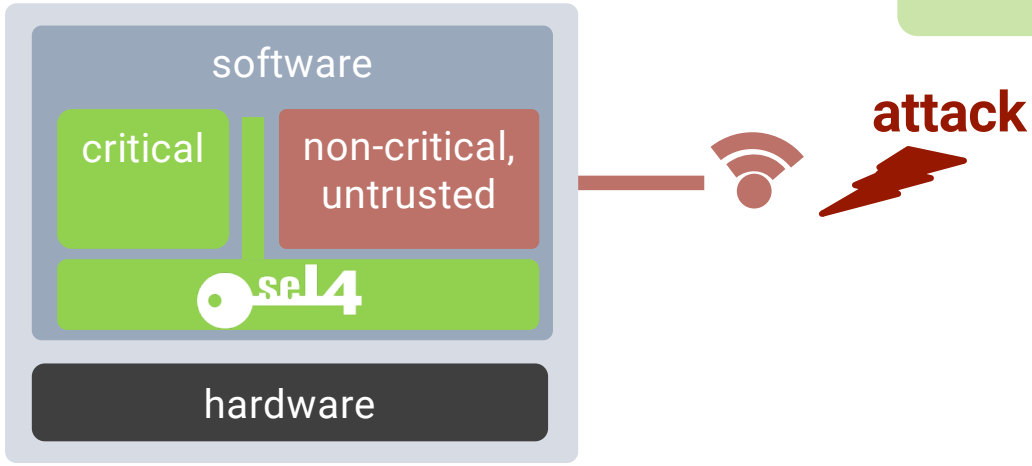
aborjának számítási logi
özhelyek nélkül komme
ni egyedülálló teljesítmé

megbízhatóságot kapnak a szoftvertől, amely e

What is seL4?

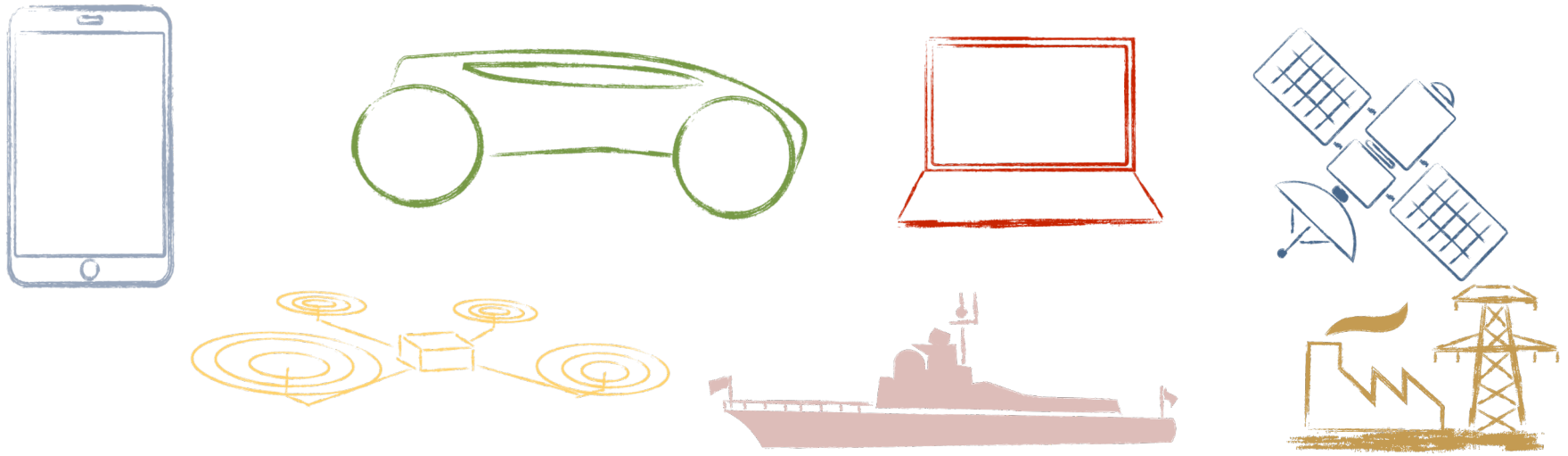
seL4 is an open source, high-assurance, high-performance operating system microkernel

- Available on GitHub under GPLv2 license
- World's most comprehensive mathematical proofs of correctness and security
- World's fastest microkernel
- Piece of software that runs at the heart of any system and controls all accesses to resources



What is seL4?

seL4 is the most trustworthy foundation for safety- and security-critical systems



Already in use across many domains:
automotive, aviation, space, defence, critical infrastructure, cyber-physical systems, IoT, industry 4.0, certified security...

The Performance Benchmark

Latency (in cycles) of a round-trip cross-address-space IPC on x64

Source	seL4	Fisco.OC	Zircon
Mi et al, 2019	986	2717	8157
Gu et al, 2020	1450	3057	8151
seL4.systems, Nov'20	797	N/A	N/A

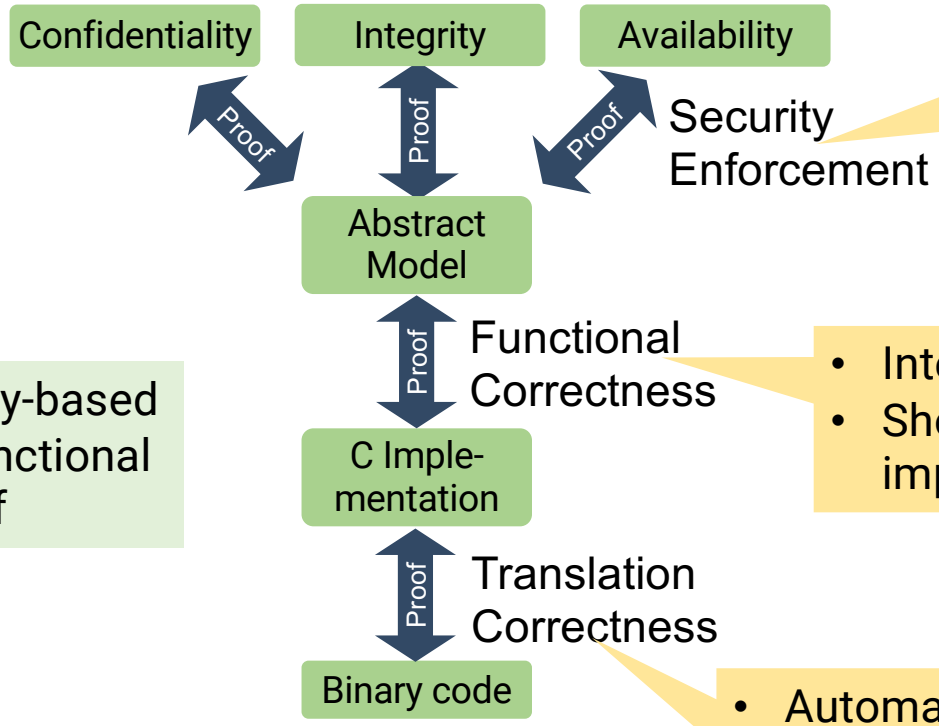
World's fastest microkernel!

Temporary performance regression in Dec'19

Sources:

- Zeyu Mi, Dingji Li, Zihan Yang, Xinran Wang, Haibo Chen: "SkyBridge: Fast and Secure Inter-Process Communication for Microkernels", EuroSys, April 2020
- Jinyu Gu, Xinyue Wu, Wentai Li, Nian Liu, Zeyu Mi, Yubin Xia, Haibo Chen: "Harmonizing Performance and Isolation in Microkernels with Efficient Intra-kernel Isolation and Communication", Usenix ATC, June 2020
- seL4 Performance, https://sel4.systems/About/Performance/, accessed 2020-11-08

Proofs



- Interactive theorem proving
- Shows kernel can guarantee security properties

Still only capability-based OS kernel with functional correctness proof

- Interactive theorem proving
- Shows C code correctly implements specification

- Automated tool chain
- Shows binary is correct translation of C

Functional Correctness Summary

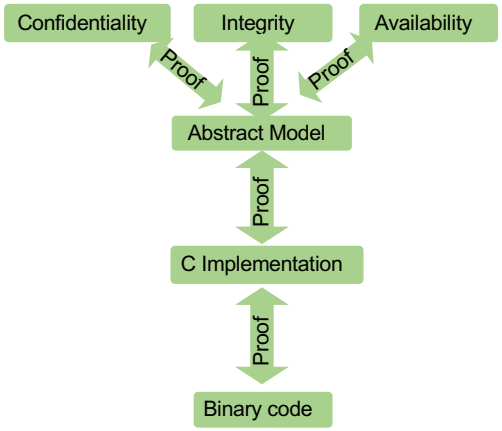
Kinds of properties proved

- Behaviour of C code is fully captured by abstract model
- Behaviour of C code is fully captured by executable model
- Kernel never fails, behaviour is always well-defined
  - assertions never fail
  - will never de-reference null pointer
  - will never access array out of bounds
  - cannot be subverted by misformed input
- All syscalls terminate, reclaiming memory is safe, ...
- Well typed references, aligned objects, kernel always mapped…
- Access control is decidable

Can prove further properties on abstract level!

Bugs found:
- 16 in (shallow) testing
- 460 in verification
  - 160 in C,
  - 150 in design,
  - 150 in spec

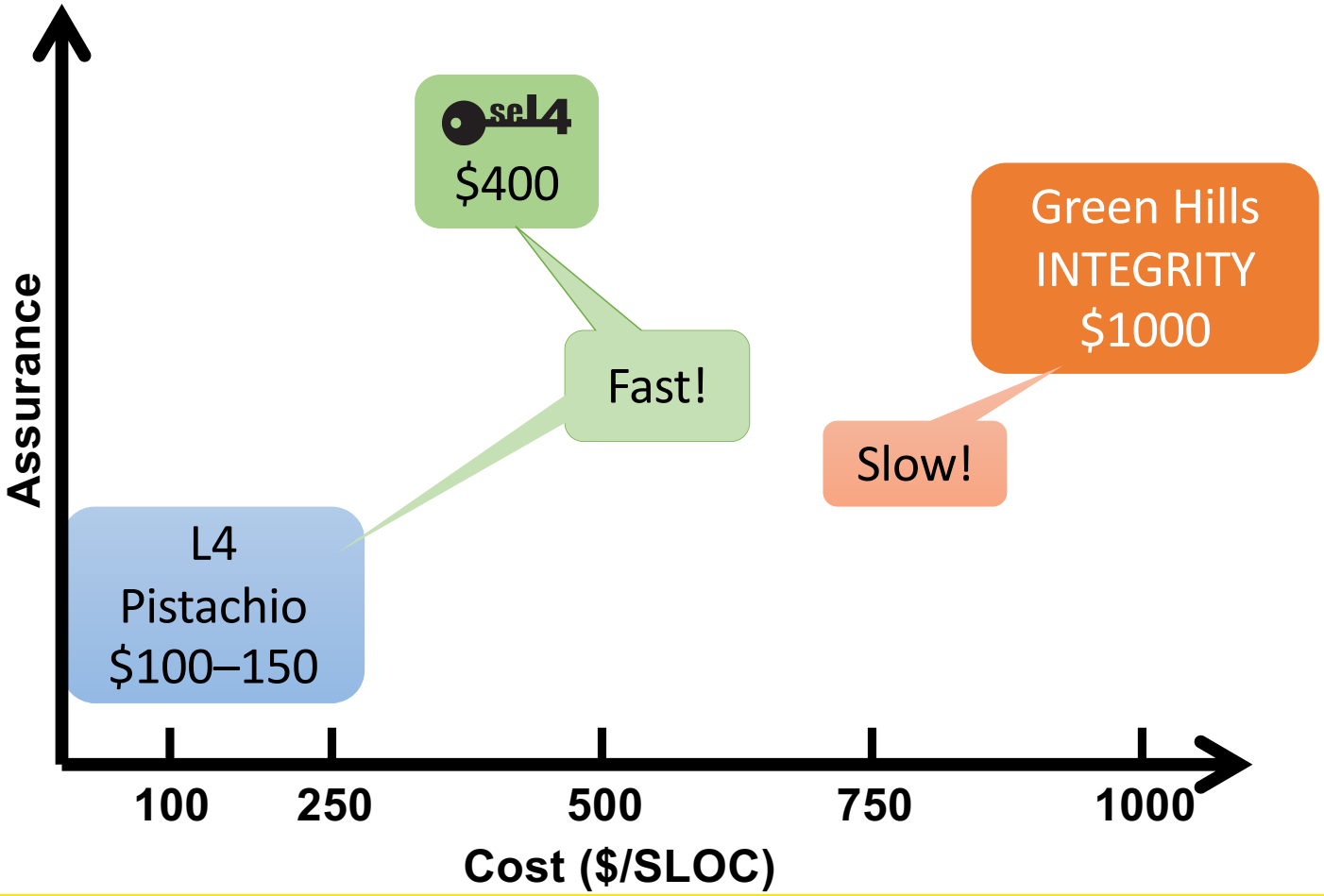
Verification Assumptions



1. Hardware behaves as expected
   - Formalised hardware-software contract (ISA)
   - Hardware implementation free of bugs, Trojans, …
2. Spec matches expectations
   - Can only prove “security” if specify what “security” means
   - Spec may not be what we think it is
3. Proof checker is correct
   - Isabel/HOL checking core that validates proofs against logic

With binary verification do **not** need to trust C compiler!

Verification Cost in Context



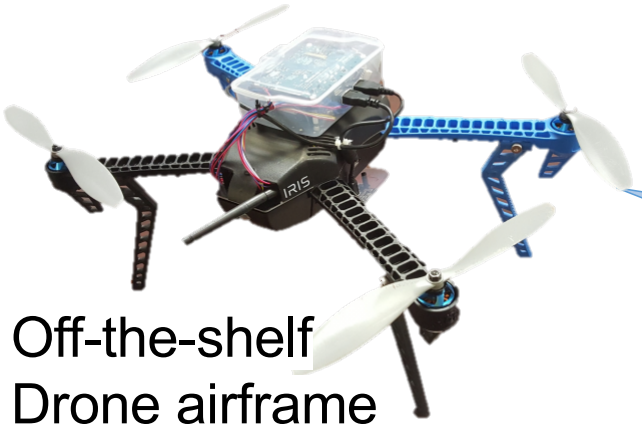
Real-World Use

seL4 DARPA HACMS

Unmanned Little Bird (ULB)

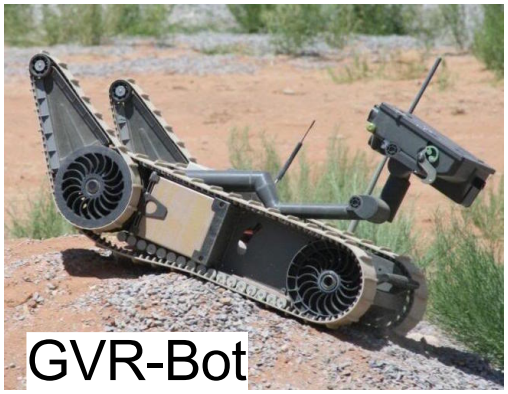
Retrofit existing system!

Autonomous trucks



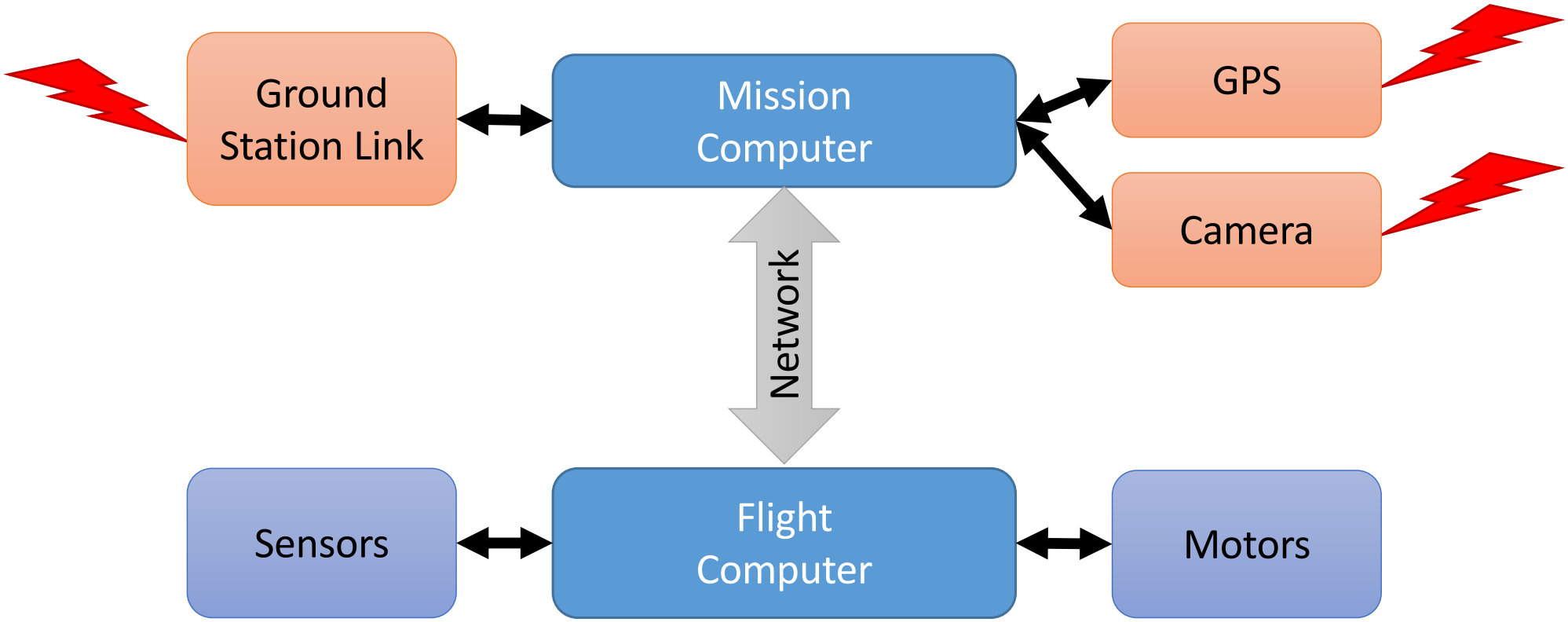
Off-the-shelf Drone airframe

Develop technology

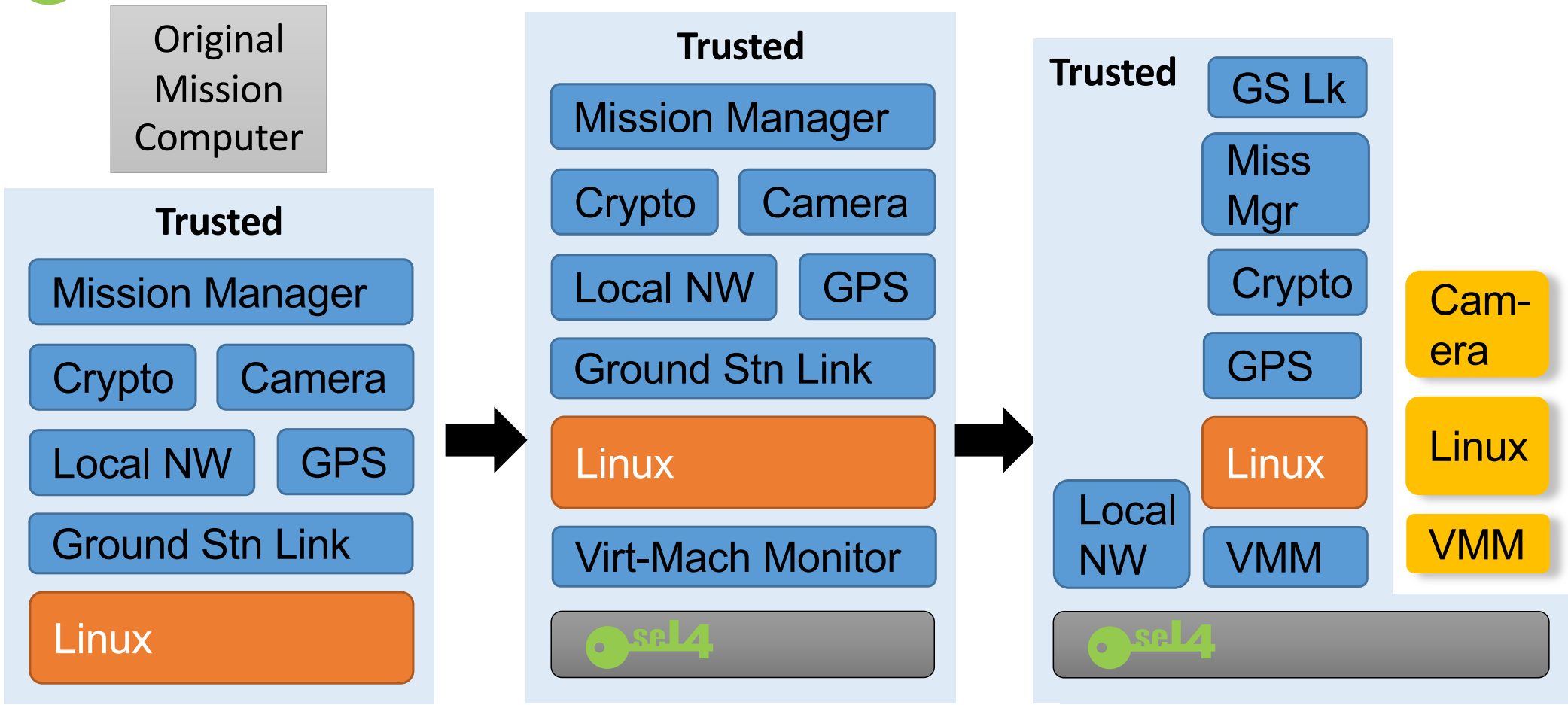


GVR-Bot

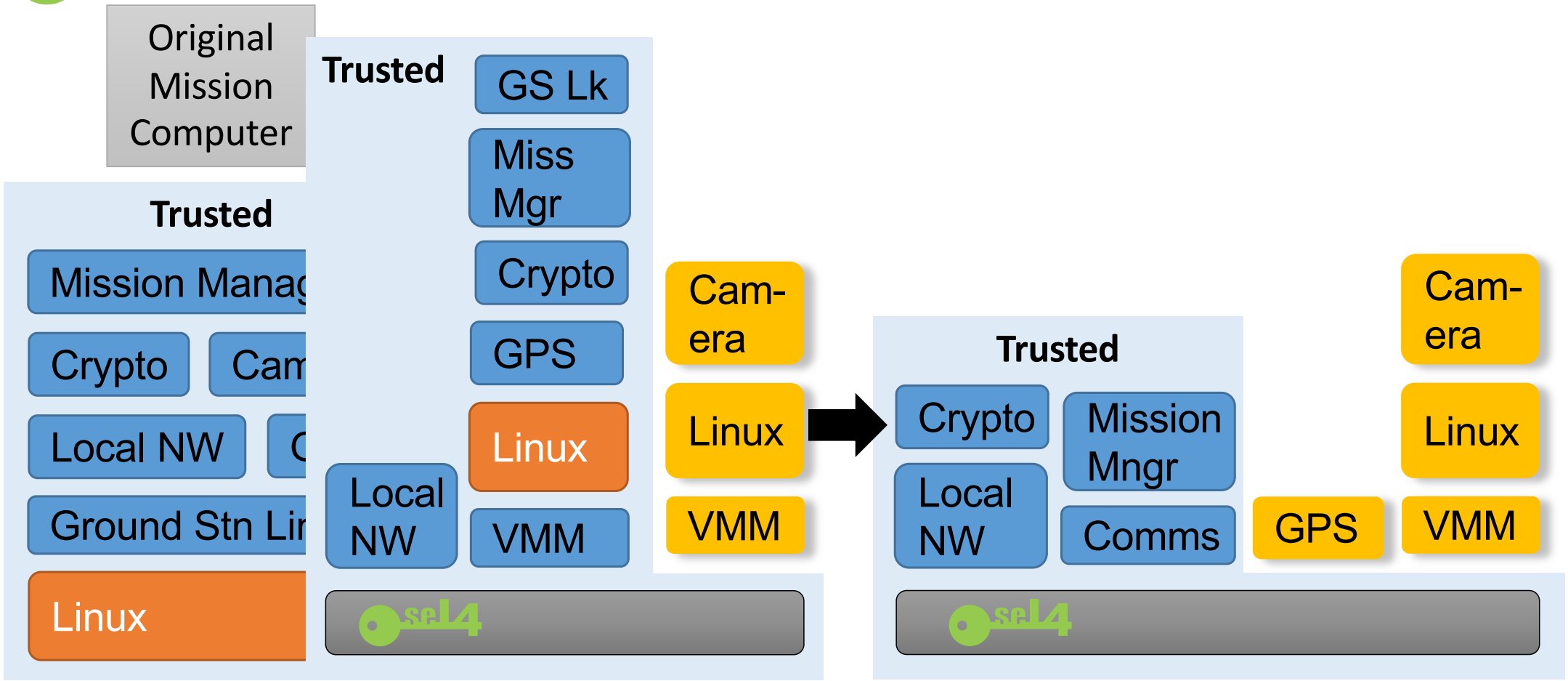
ULB Architecture



Incremental Cyber Retrofit



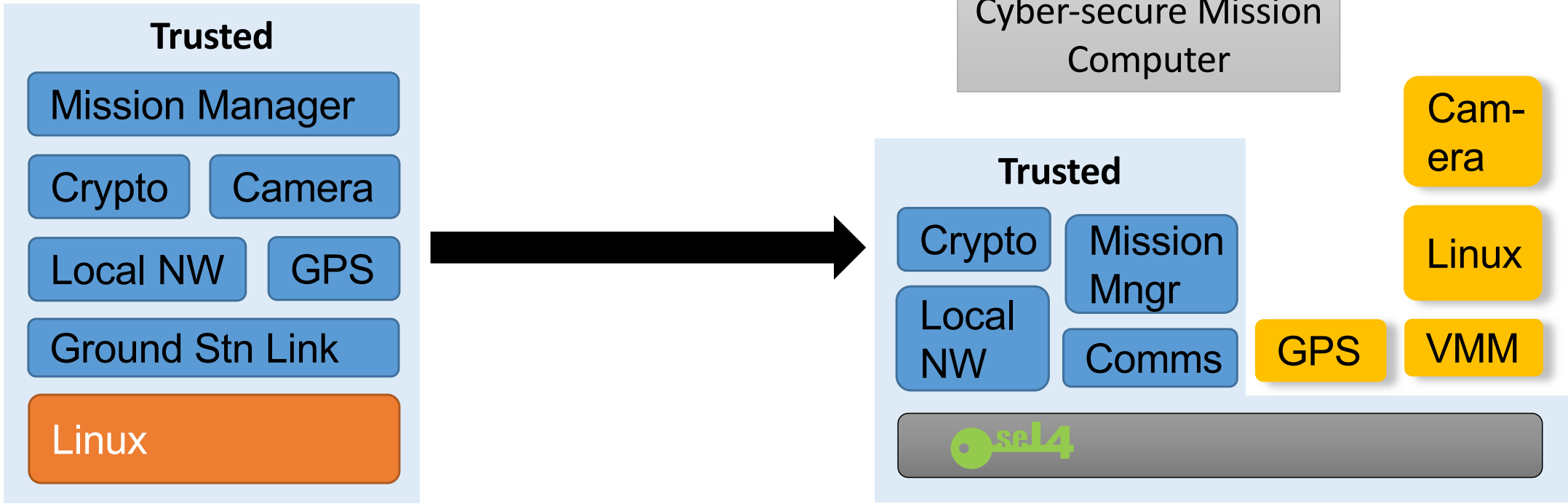
Incremental Cyber Retrofit



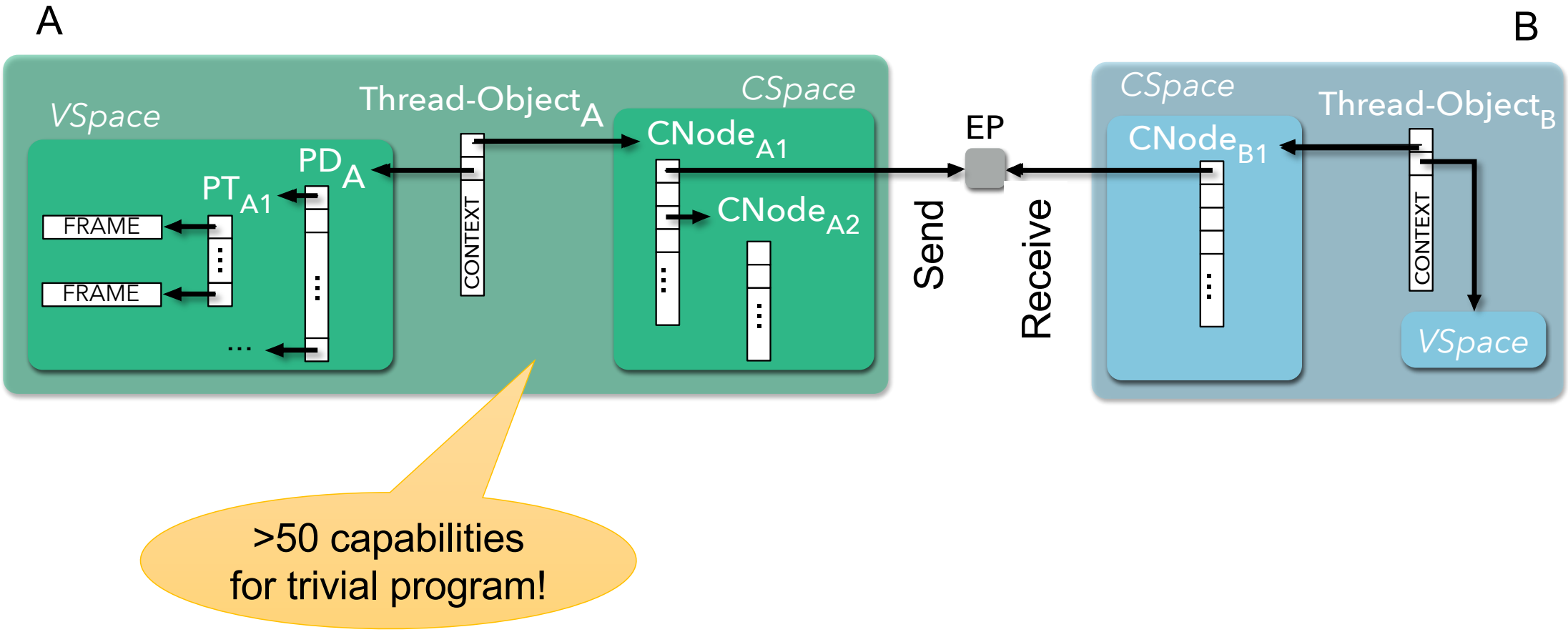
Incremental Cyber Retrofit

Original Mission Computer

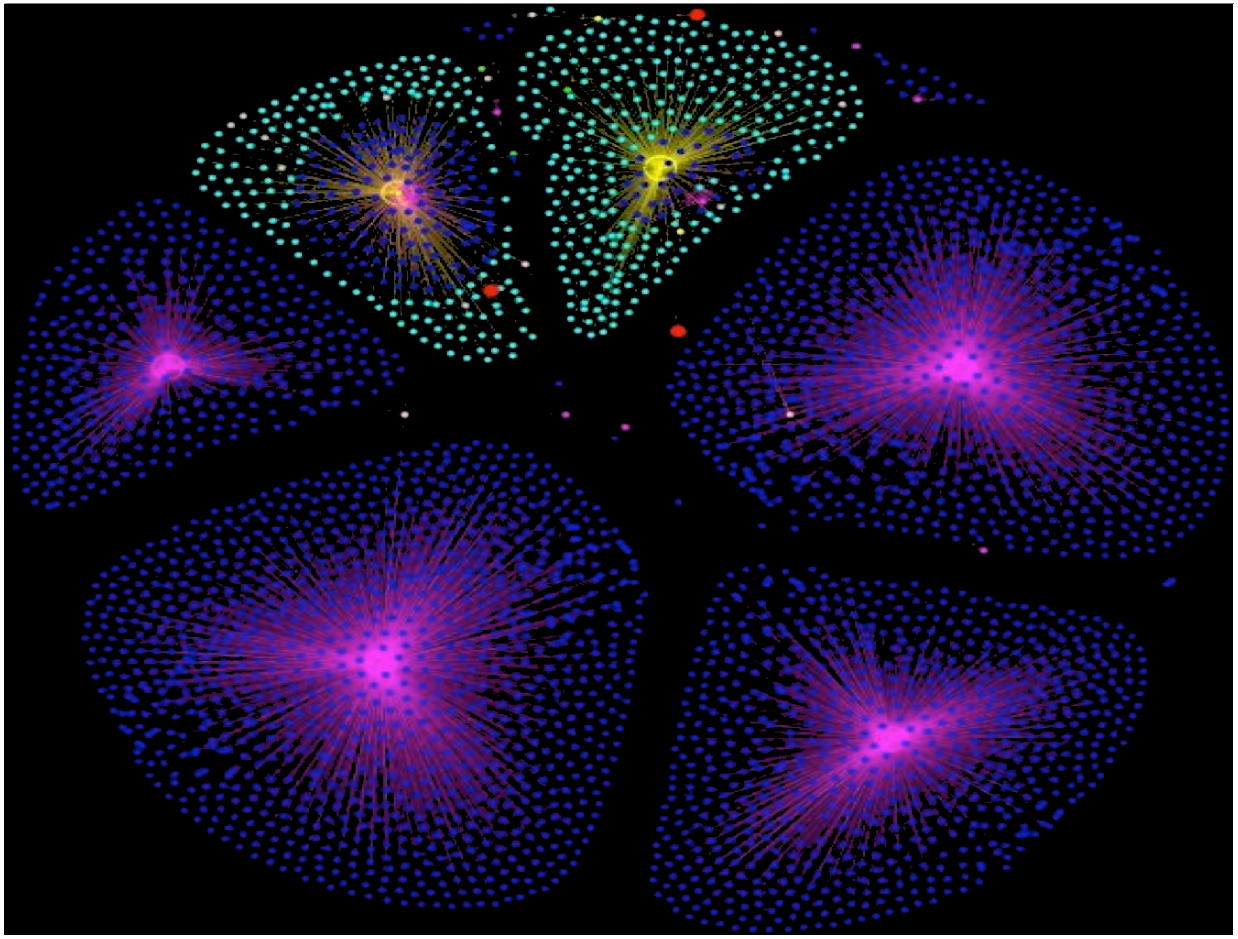
[Klein et al, CACM, Oct'18]



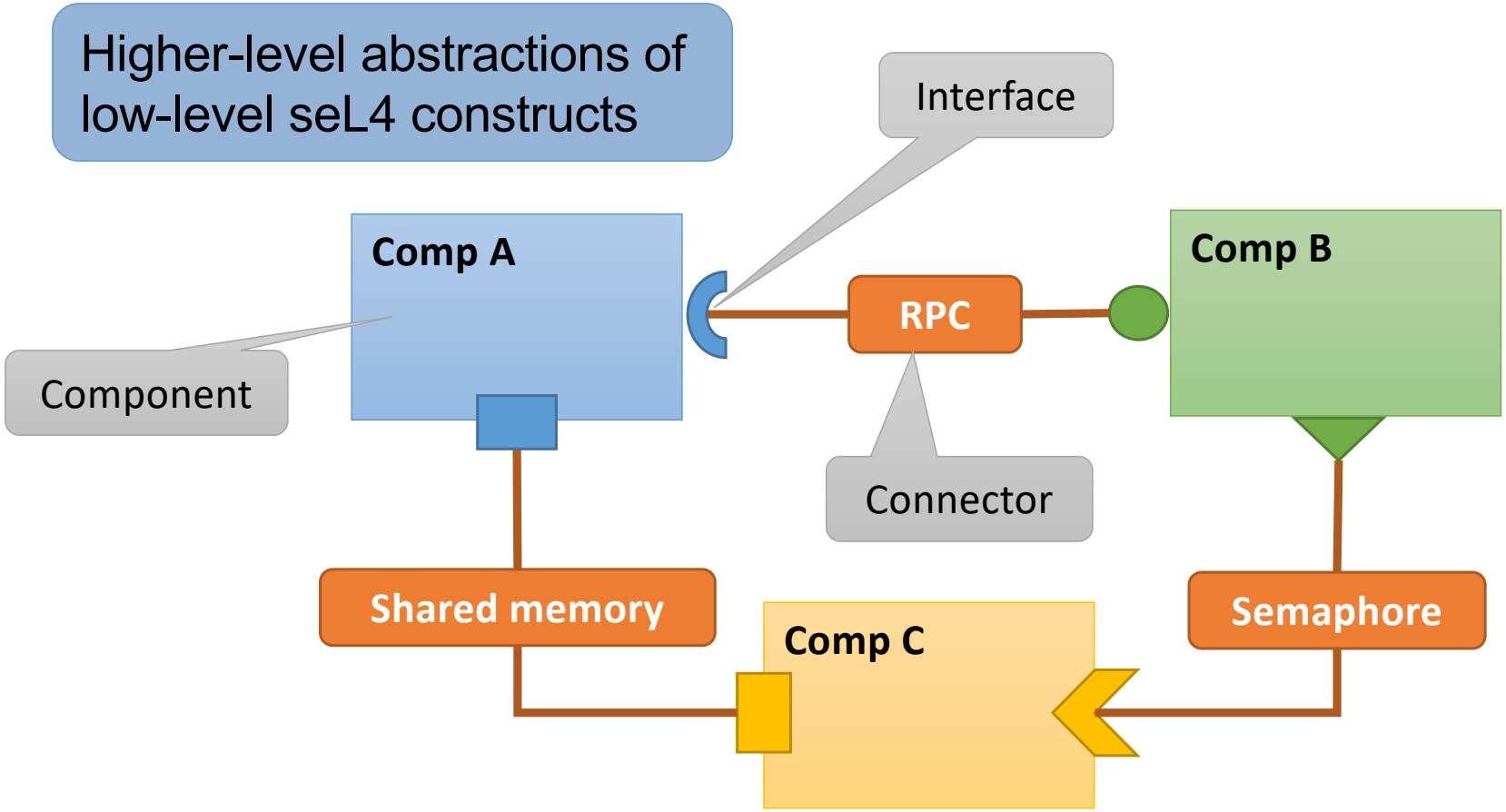
Issue: seL4 Objects are Low-Level



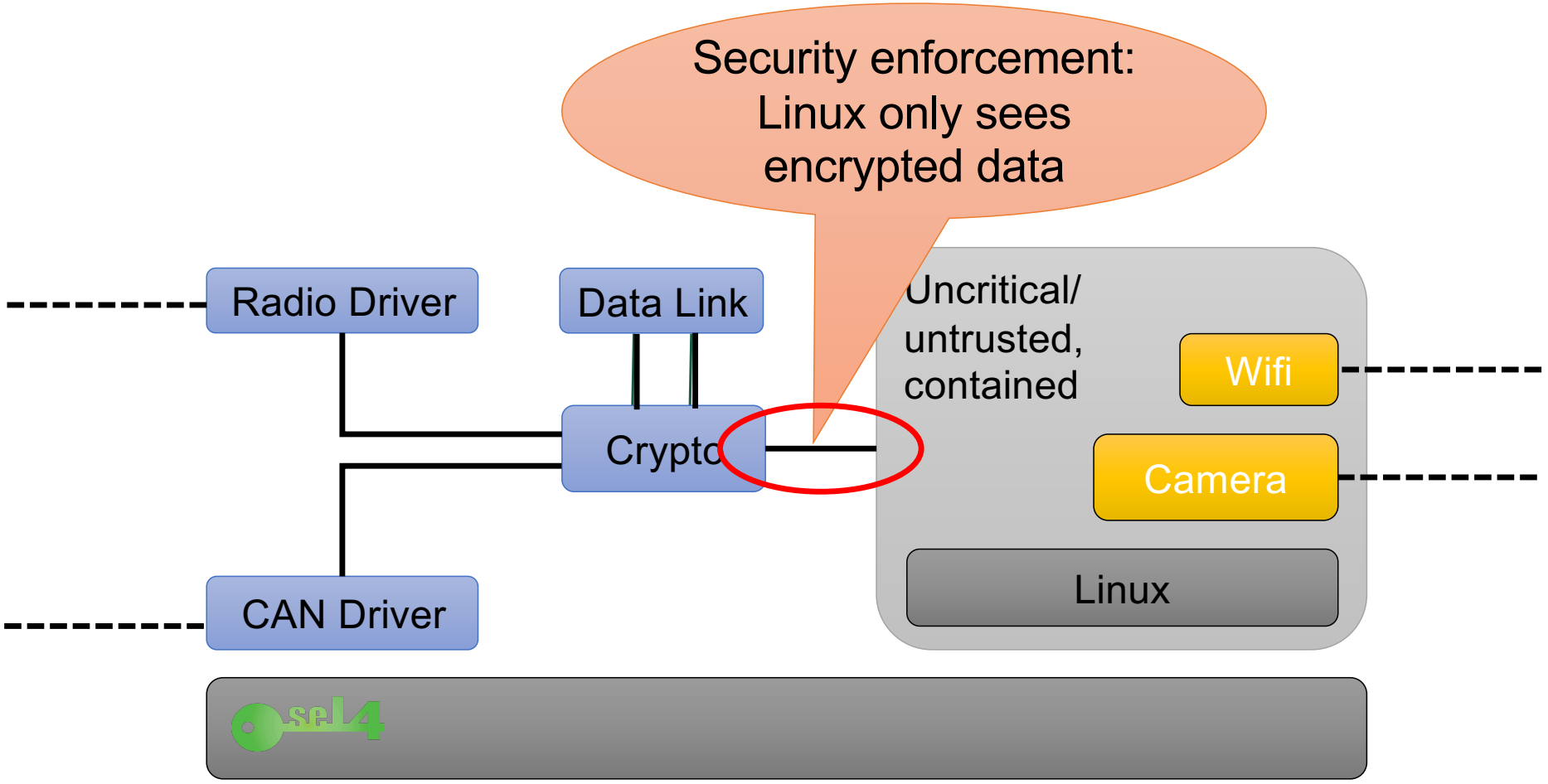
Simple But Non-Trivial System



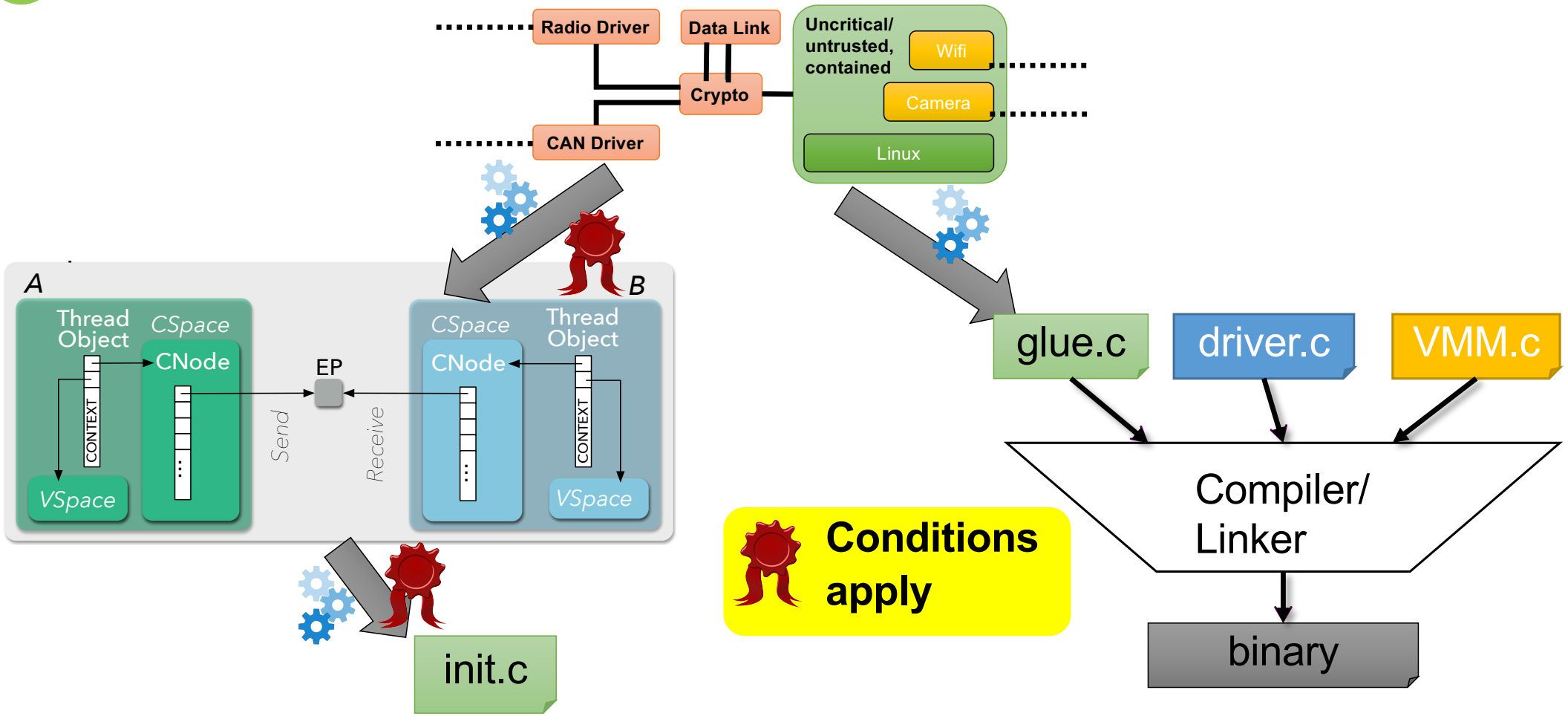
Component Middleware: CAmkES



HACMS UAV Architecture



Enforcing the Architecture



Military-Grade Security

Cross-Domain Desktop Compositor

Multi-level secure terminal

- Successful trials in AU, US, UK, CA
- Commercialisation in progress

Secure communication device in use in AU, UK defence forces

Real-World Use
Courtesy Boeing, DARPA