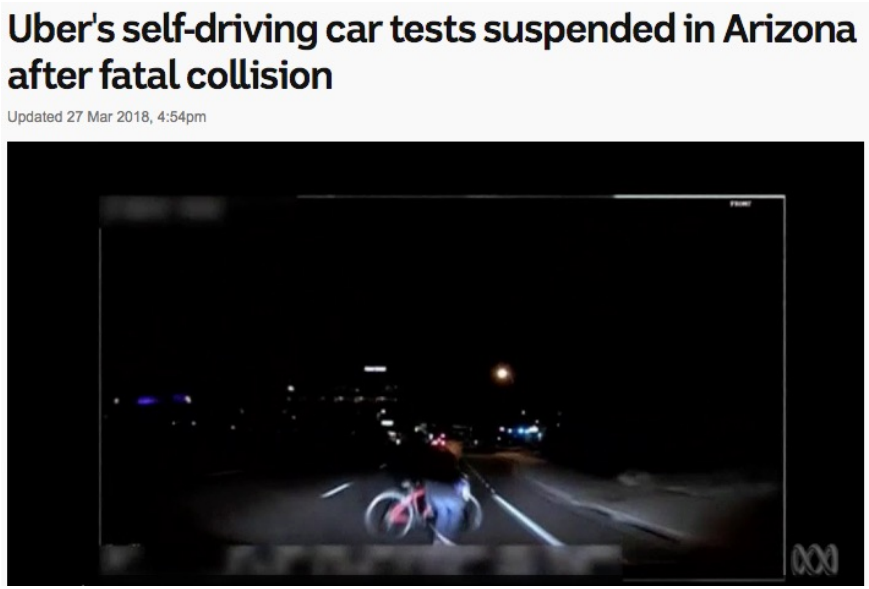
No Safety without Security, No Security without Trustworthy OS

Gernot Heiser | gernot.heiser@data61.csiro.au | @GernotHeiser
Trustworthy Systems | Data61 & UNSW Sydney

https://trustworthy.systems

CSIRO

Autonomous Car Safety

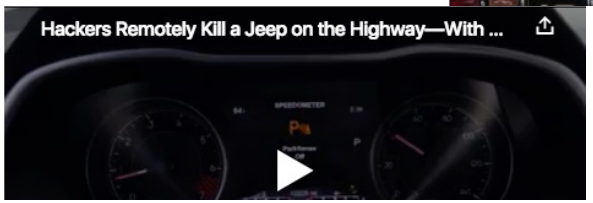


Car Security (and Implications)

Traffic chaos from Sydney Harbour Bridge drama cost city up to $10 million

ANDY GREENBERG SECURITY 07.21.15 06:00 AM

HACKERS REMOTELY
A JEEP ON THE HIGH
—WITH ME IN IT



GREENBERG SECURITY 08.16.17 04:55 PM

DEEP FLAW IN YOUR
R LETS HACKERS SHUT
WN SAFETY FEATURES

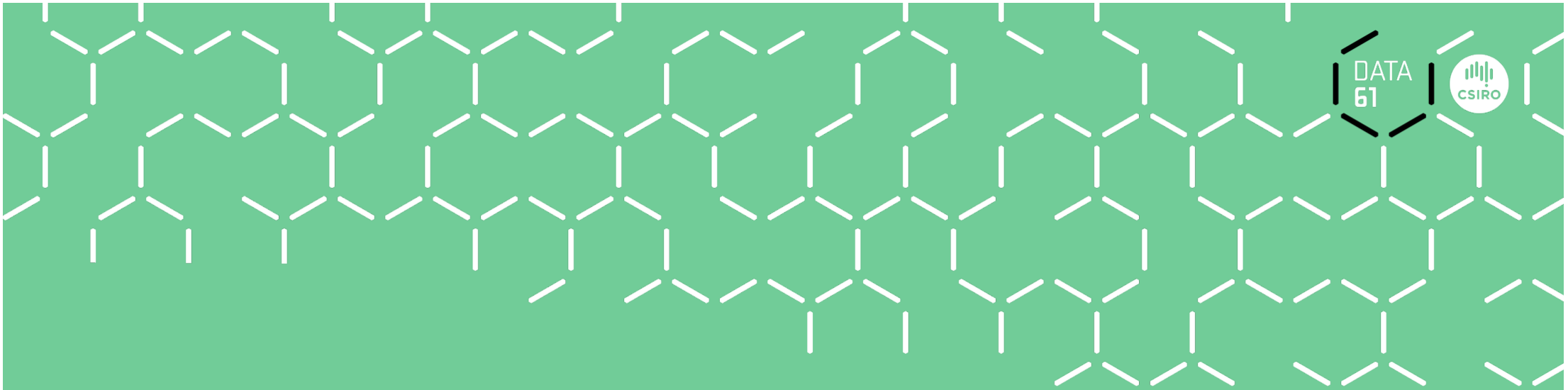
Cybersecurity: 1st Class Safety Issue

Fundamental rules of cyber space:

1. The internet is a hostile environment
2. Anything that is internet-connected *can* be attacked
3. Anything that *can* be attacked *will* be attacked

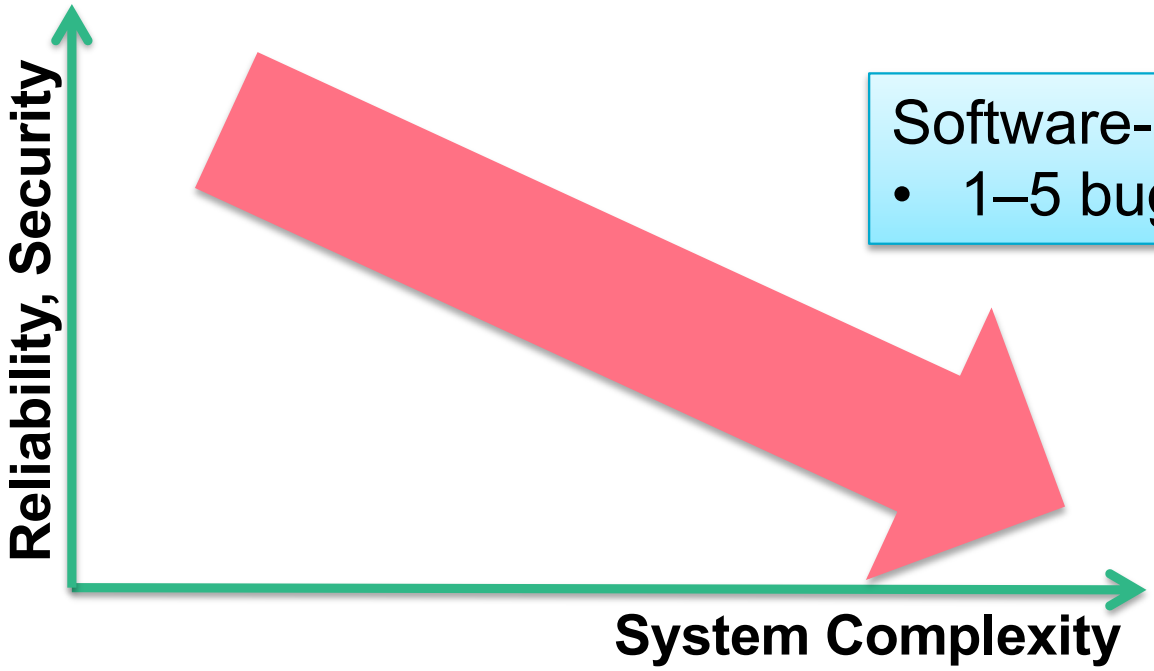
Examples:

- Cars, especially autonomous
- Trains
- Aircraft
- Robots
- Smart City infrastructure



Why Are Systems So Vulnerable?

Failure Reason #1: Complexity



Software-engineering rule of thumb:
- 1–5 bugs per 1,000 lines of ***quality*** code

Bluetooth protocol stack:
Multiple 100,000 lines

Linux/Windows OS:
Tens of millions lines

Complexity Drivers
- Features/functionality
- Legacy reuse

Failure Reason #2: Care Factor

Developer priorities

1. Features/functionality
2. Cost
3. Time to market
4. …
5. ...
6. ...
7. ...

....

....

999. Security

Developer expertise

1. Undergraduate programming
2. Application domain
3. Maybe hardware
4. …
5. ...
6. ...
7. ...

....

....

999. Security

Failure Reason #3: Security ≠ Safety

Classic safety thinking (eg automotive, electrical):

- Failures are *random*
- Failure rates can be kept *very low* through systematic process
- Multiple failures are *independent*

Software security weaknesses:

- Failure is *deterministic*
- Failure rates are *high*
- Attackers *systematically combine* multiple vulnerabilities

No safety without cyber security!

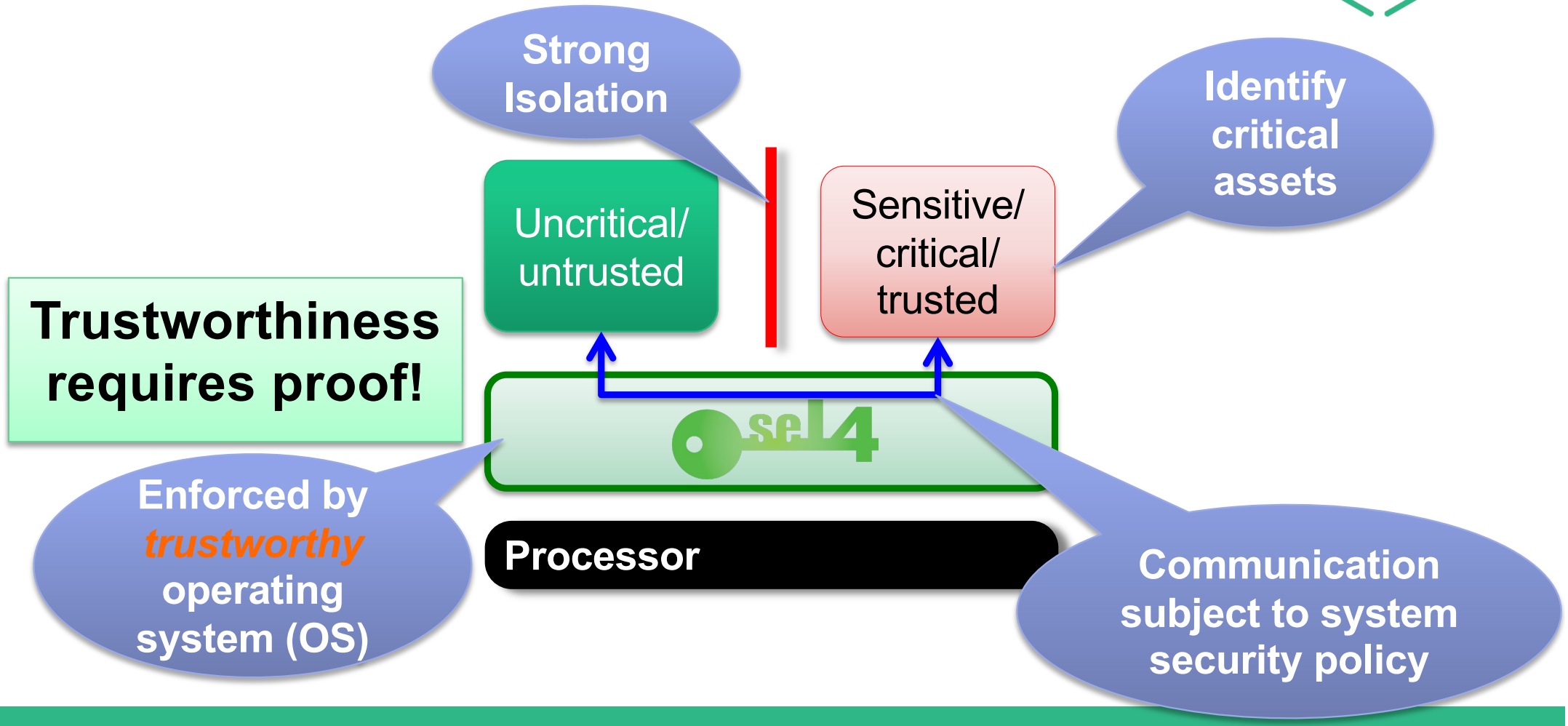
⇒ **Classical safety approaches do not work against cyber attacks!**

Standard IT “Security” Approaches Fail

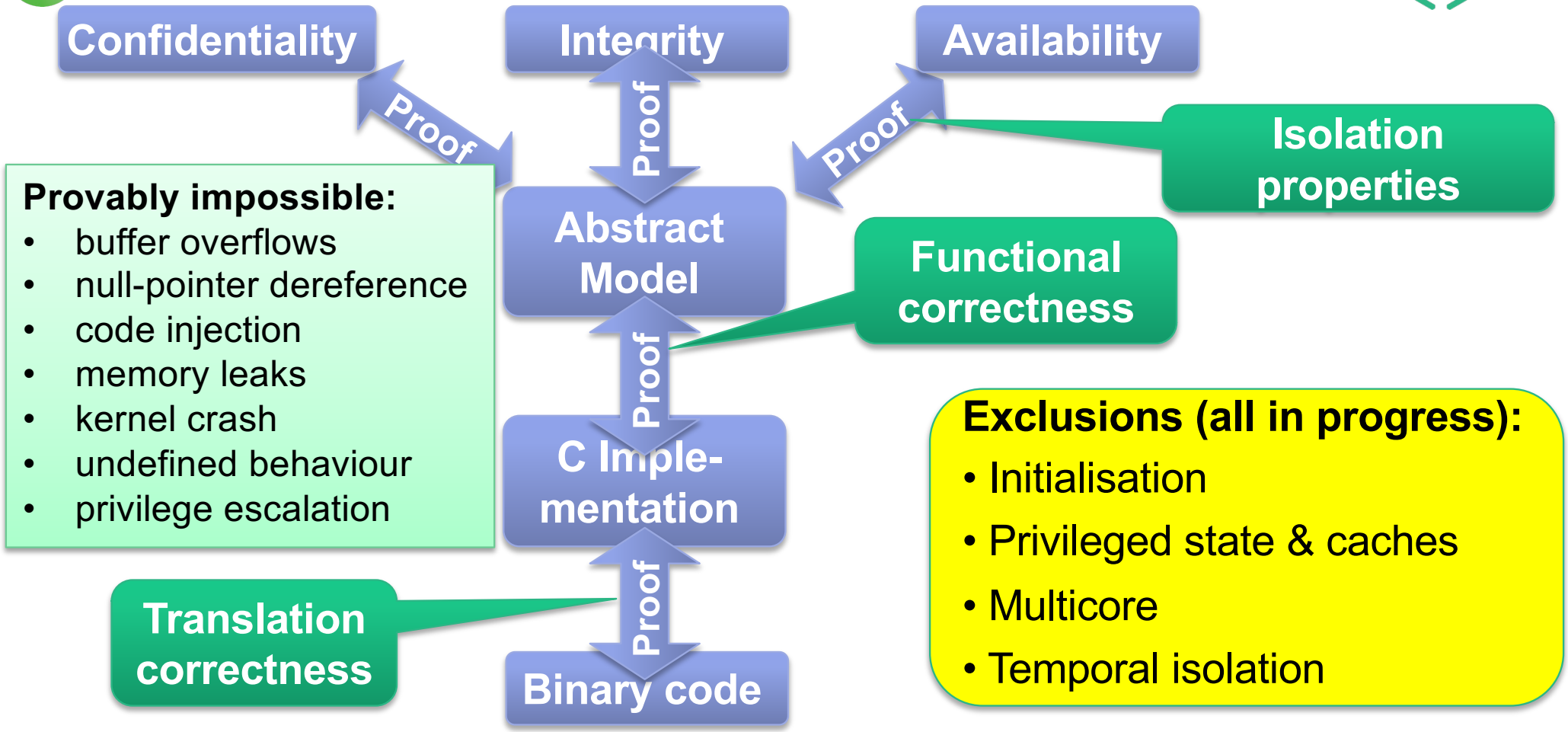
- Identify and fix vulnerabilities, aka Patch-and-pray:
  - Reactive, can only deal with attacks after they happened
- Firewalls:
  - Run on potentially compromised operating system
  - Cannot protect against compromised traffic from authorised source
- Machine-learning based intrusion detection
  - Reactive, assumes system is already compromised
  - Runs on potentially compromised operating system
  - Admission of defeat

What Is Needed?

Core Security Requirement: Isolation



We Have Proof!



Military-Grade Security

US Army Autonomous Trucks

Crypto Stick

Boeing Unmanned Helicopter

Cross-Domain Desktop Compositor

Summary

- Autonomous cyber-physical systems are highly vulnerable
- Classical defences don’t work (even less so than in enterprise IT)
- Real security can only be achieved by
  - Security-oriented system architecture
  - Rock-solid operating-system foundation

Good news: A real solution exists!

Thank you

Gernot Heiser | gernot.heiser@data61.csiro.au | @GernotHeiser

https://trustworthy.systems

CSIRO