November 21, 2024

The CHERI Alliance

C&ESAR Conference

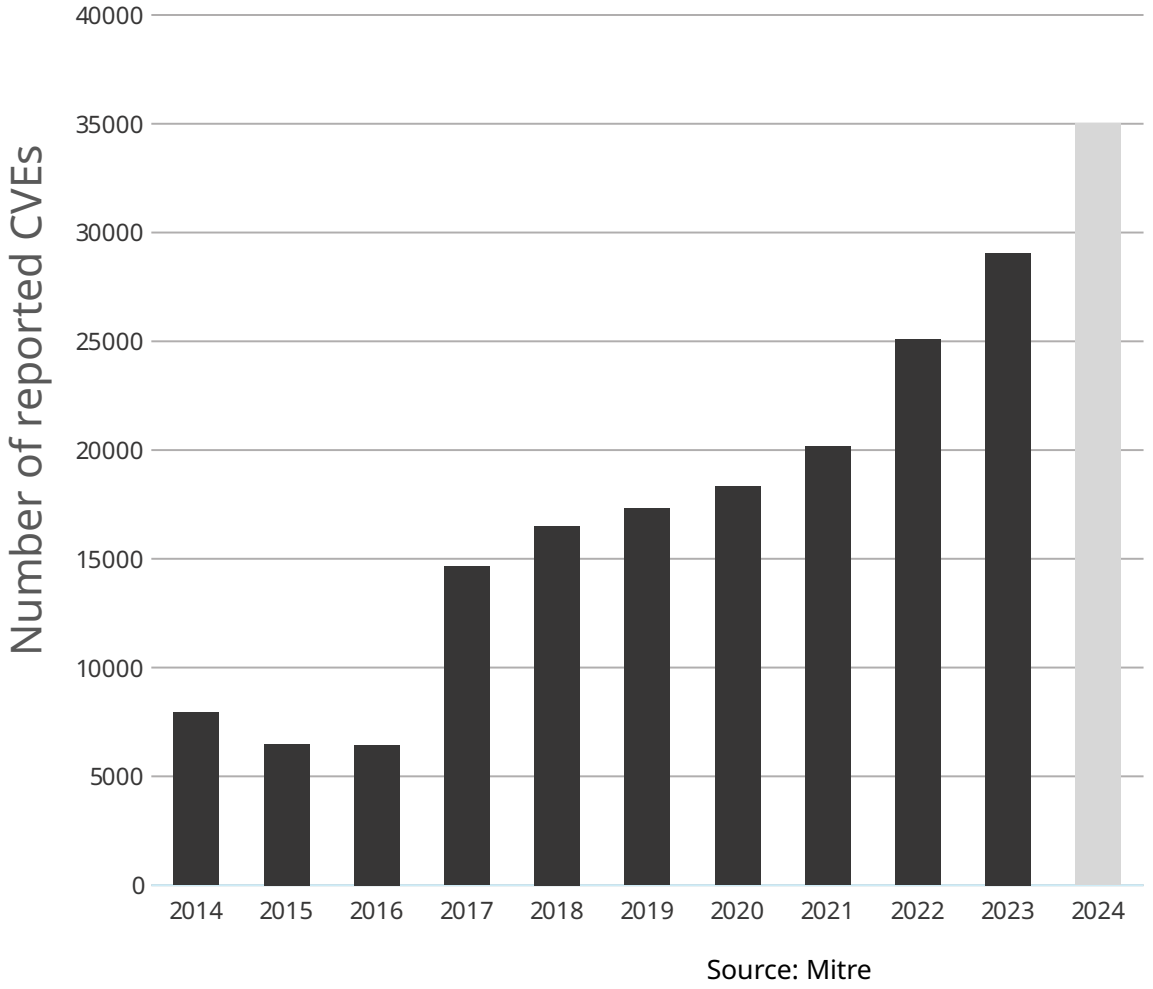
Mike EFTIMAKIS
Founding Director – CHERI Alliance

Agenda

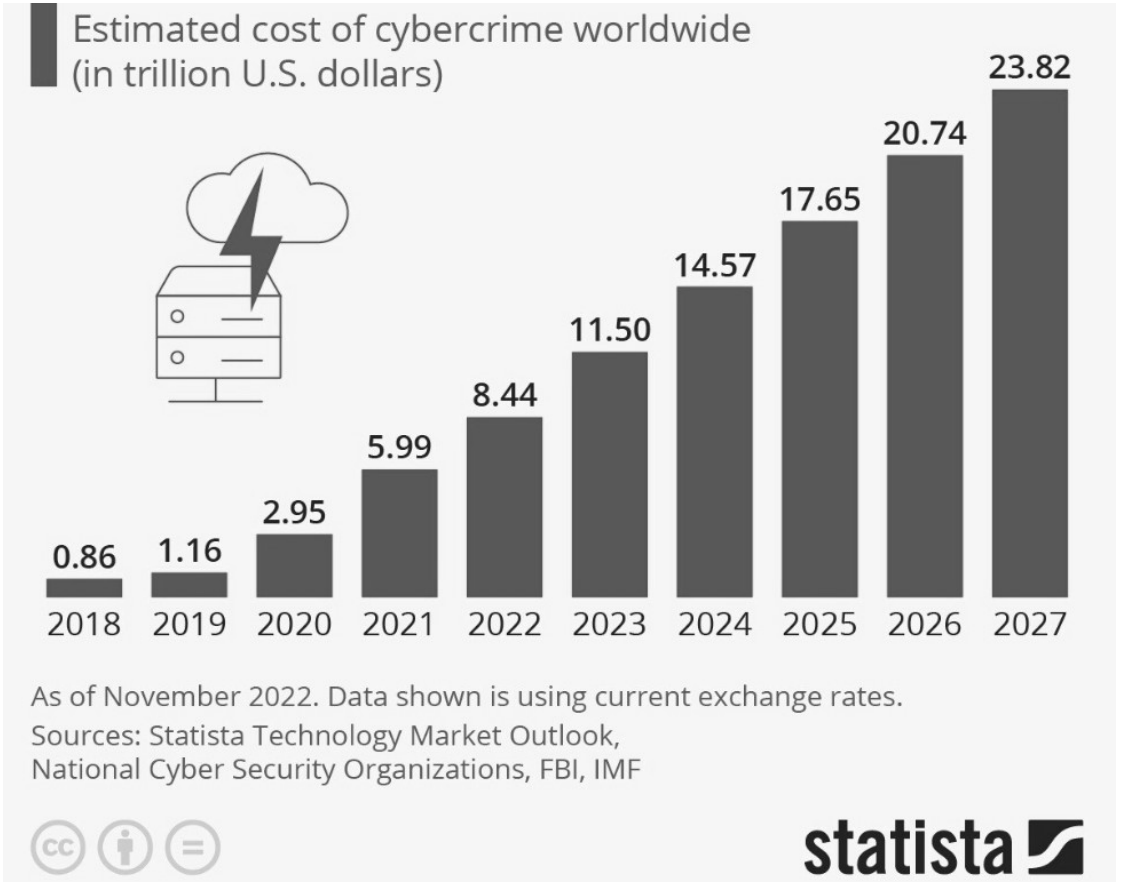
- The memory safety problem
- CHERI technology
- The CHERI Alliance

The memory safety problem

Vulnerabilities are causing increasing risk

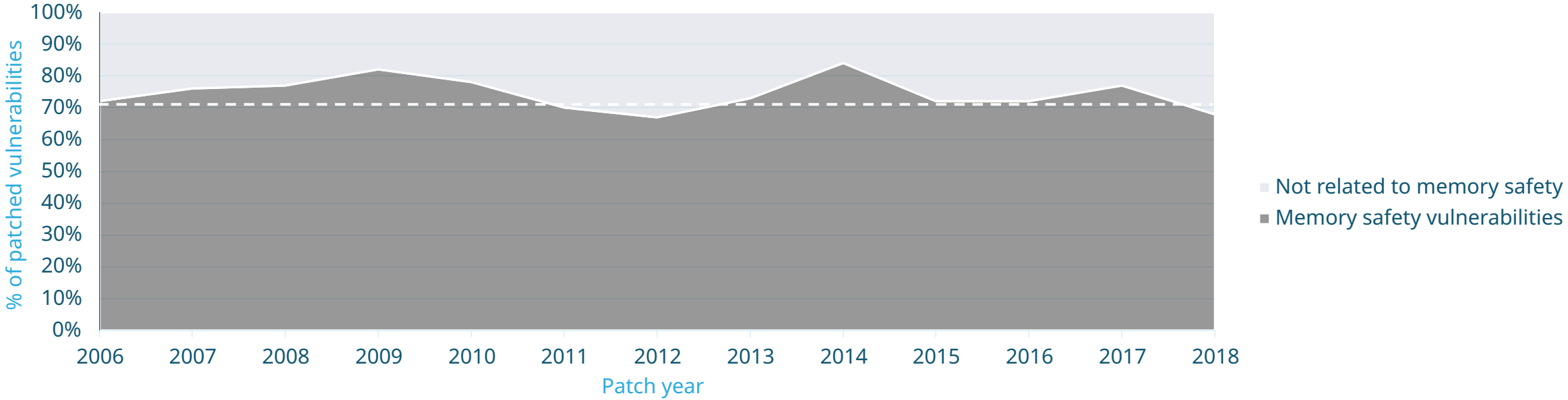


Source: Mitre



The main problem is (the lack of) memory safety

- Memory abuse (e.g. buffer overflows) is the main attack vector
- Constant ratio of over the past 20 years
  - … even with all the work done on software to avoid this!

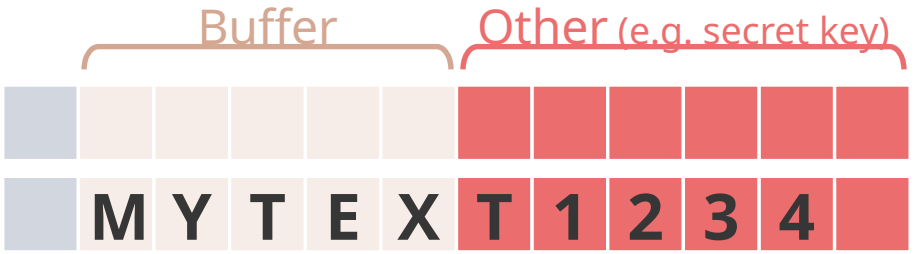


Source: Trends, challenge, and shifts in software vulnerability mitigation - Microsoft 2019

Example of memory safety issue

Buffer overflow

- Storing text in memory
  - If the allocated space is too small, then
  - Text overwrites other data



Simple but...

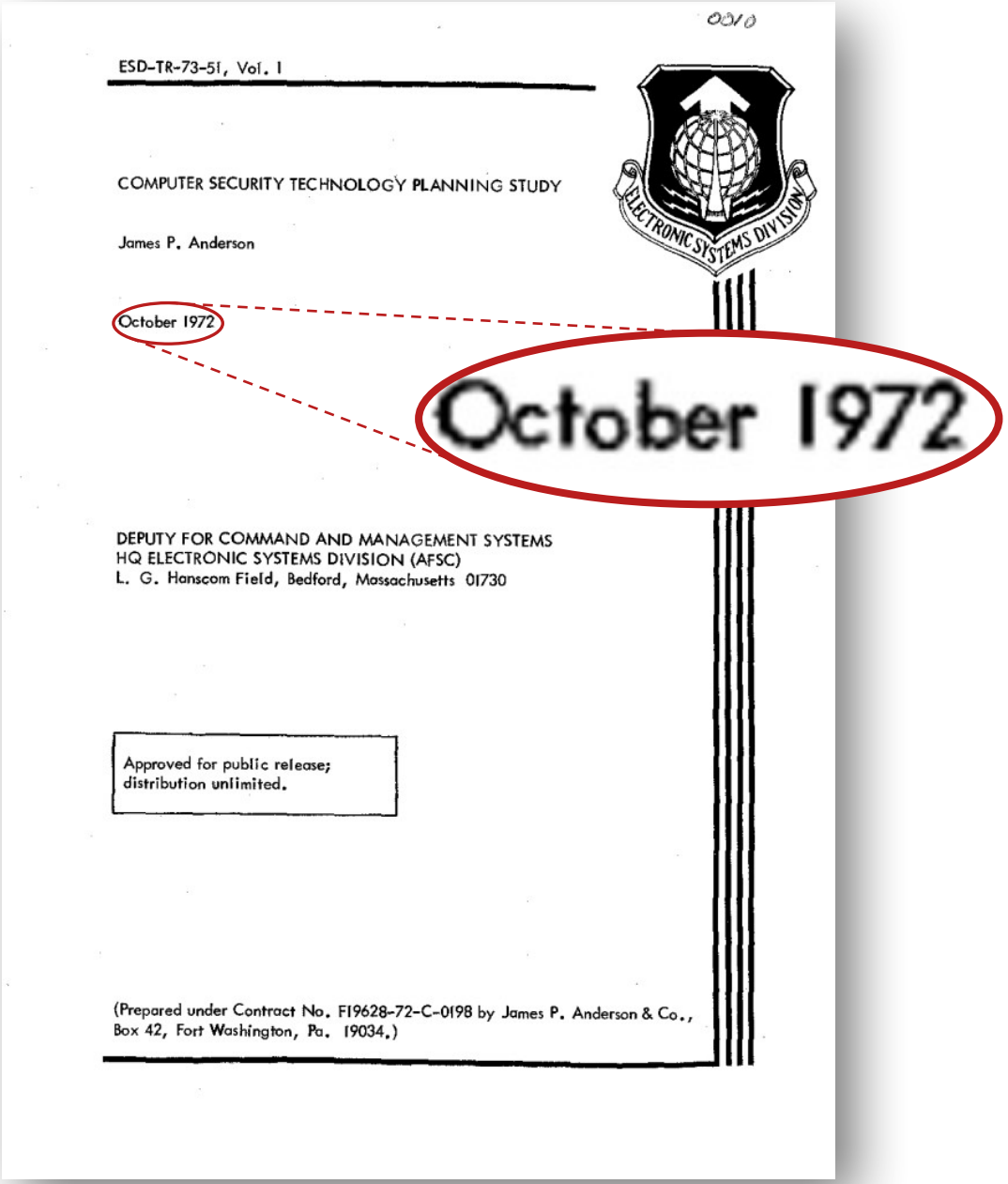
Attack	Cost
Morris Worm	$100,000 to $10 million
Heartbleed	$500 million
Code Red Worm	$2.6 billion
WannaCry Ransomware	$4 billion

It’s not a new problem…

"patching" of known faults […] without any better technical foundation […] is futile for achieving multilevel security.

Unless security is designed into a system from its inception, there is little chance that it can be made secure by retrofit.

Computer Security Technology Planning Study - USAF
https://csrc.nist.gov/csrc/media/publications/conference-paper/1998/10/08/proceedings-of-the-21st-nissc-1998/documents/early-cs-papers/ande72a.pdf


0010

ESD-TR-73-51, Vol. I

COMPUTER SECURITY TECHNOLOGY PLANNING STUDY

James P. Anderson

October 1972

October 1972

DEPUTY FOR COMMAND AND MANAGEMENT SYSTEMS
HQ ELECTRONIC SYSTEMS DIVISION (AFSC)
L. G. Hanscom Field, Bedford, Massachusetts 01730

Approved for public release;
distribution unlimited.

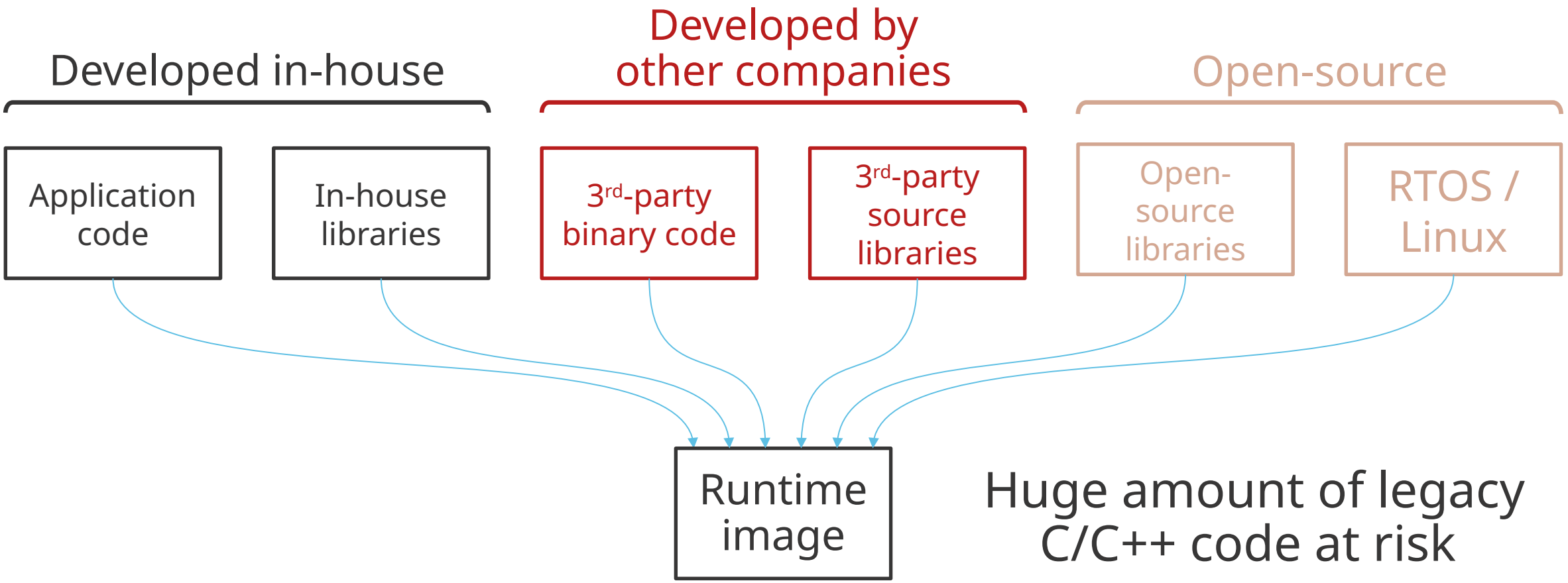
(Prepared under Contract No. F19628-72-C-0198 by James P. Anderson & Co., Box 42, Fort Washington, Pa. 19034.)

Possible solutions for memory safety

Possible solutions for memory safety

Use “memory safe” languages like Rust or .Net ?

⬡ Impossible to re-write software to fix the problem



Possible solutions for memory safety

✗ Use “memory safe” languages like Rust or .Net

- Requires rewriting trillions of lines of C/C++ code
- Possible for new code, but no compartmentalisation

Possible solutions for memory safety

✗ Use “memory safe” languages like Rust or .Net

- Requires rewriting trillions of lines of C/C++ code
- Possible for new code, but no compartmentalisation

Use “coarse-grained” techniques like stack “canaries”

Possible solutions for memory safety

- ✗ Use “memory safe” languages like Rust or .Net
  - Requires rewriting trillions of lines of C/C++ code
  - Possible for new code, but no compartmentalisation
- ✗ Use “coarse-grained” techniques like stack “canaries”
  - Helpful, but they statistically leave too many holes
  - Hacking techniques already developed

Possible solutions for memory safety

✗ Use “memory safe” languages like Rust or .Net

- Requires rewriting trillions of lines of C/C++ code
- Possible for new code, but no compartmentalisation

✗ Use “coarse-grained” techniques like stack “canaries”

- Helpful, but they statistically leave too many holes
- Hacking techniques already developed

Use “fine-grained” techniques like CHERI

Possible solutions for memory safety

✗ Use “memory safe” languages like Rust or .Net

- Requires rewriting trillions of lines of C/C++ code
- Possible for new code, but no compartmentalisation

✗ Use “coarse-grained” techniques like stack “canaries”

- Helpful, but they statistically leave too many holes
- Hacking techniques already developed

✓ Use “fine-grained” techniques like CHERI

- Best option, but needs new hardware

CHERI technology

Capability **H**ardware **E**nhanced **R**ISC

About CHERI

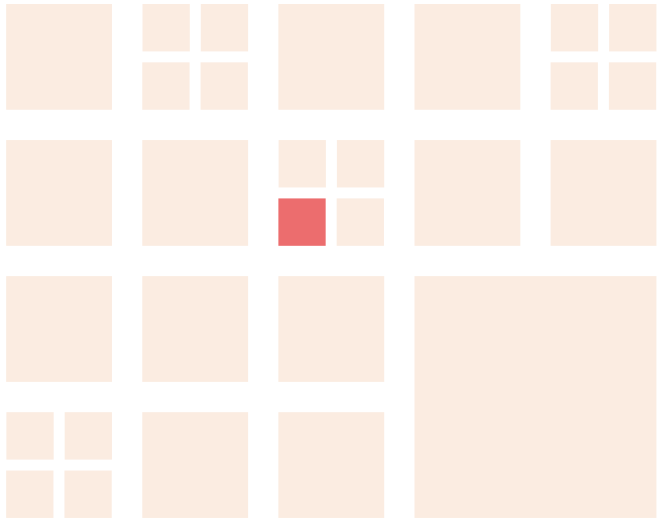
- Initiated by a project from

- Originally developed by

- Matured for 14 years
- Supported by

CHERI

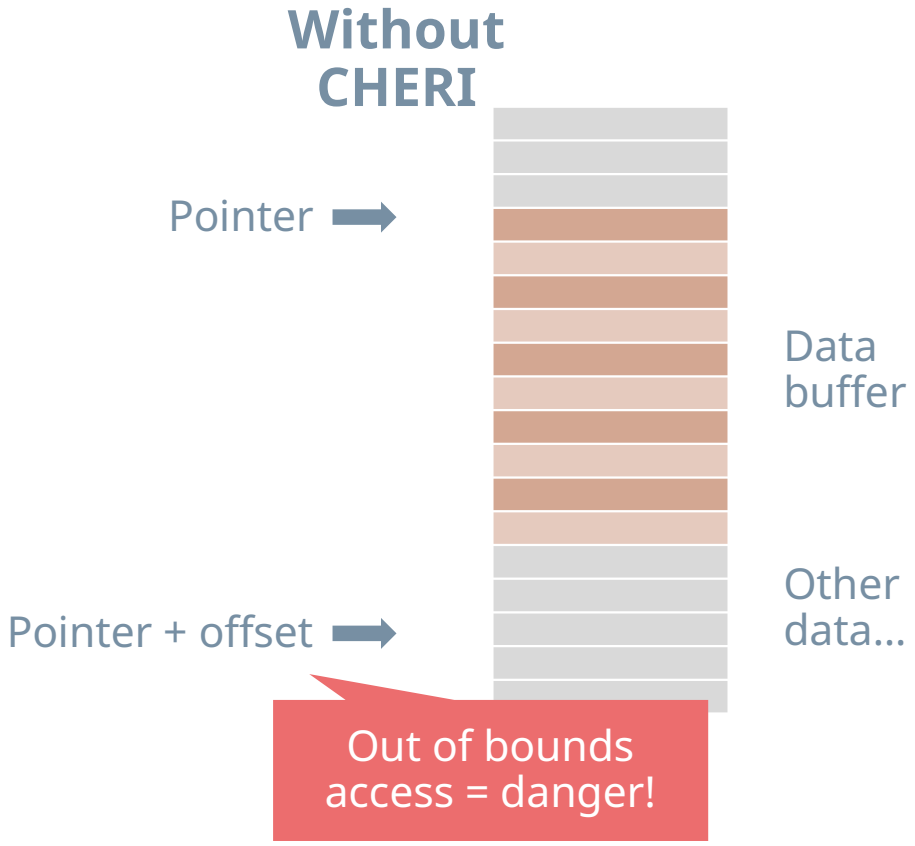
- Fine-grained **memory protection**
  - Hardware enforced
- **Compartmentalization**
  - Principle of least privilege
- Formally proven protection

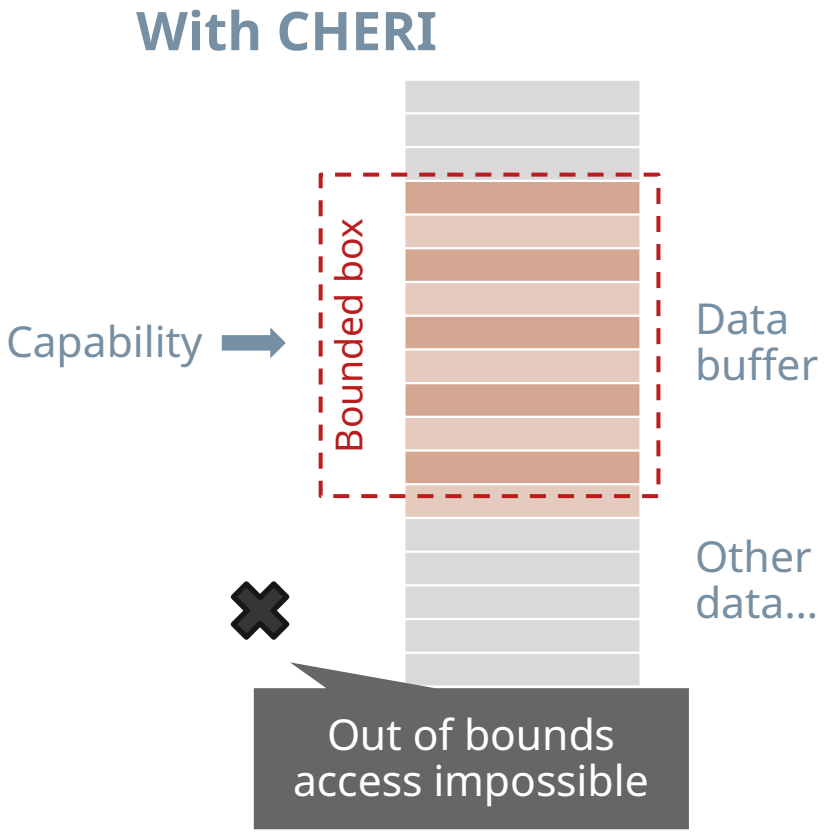


Compartmentalization prevents contagion

Spatial memory safety

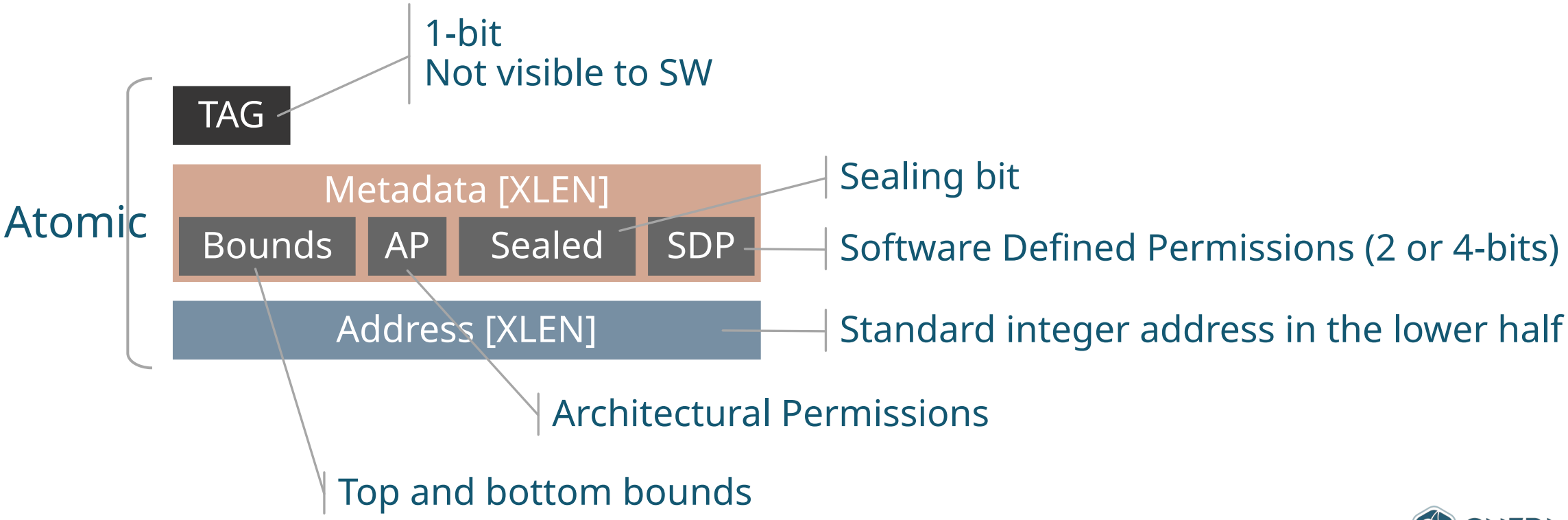
- Replacing pointers by capabilities – with hardware control





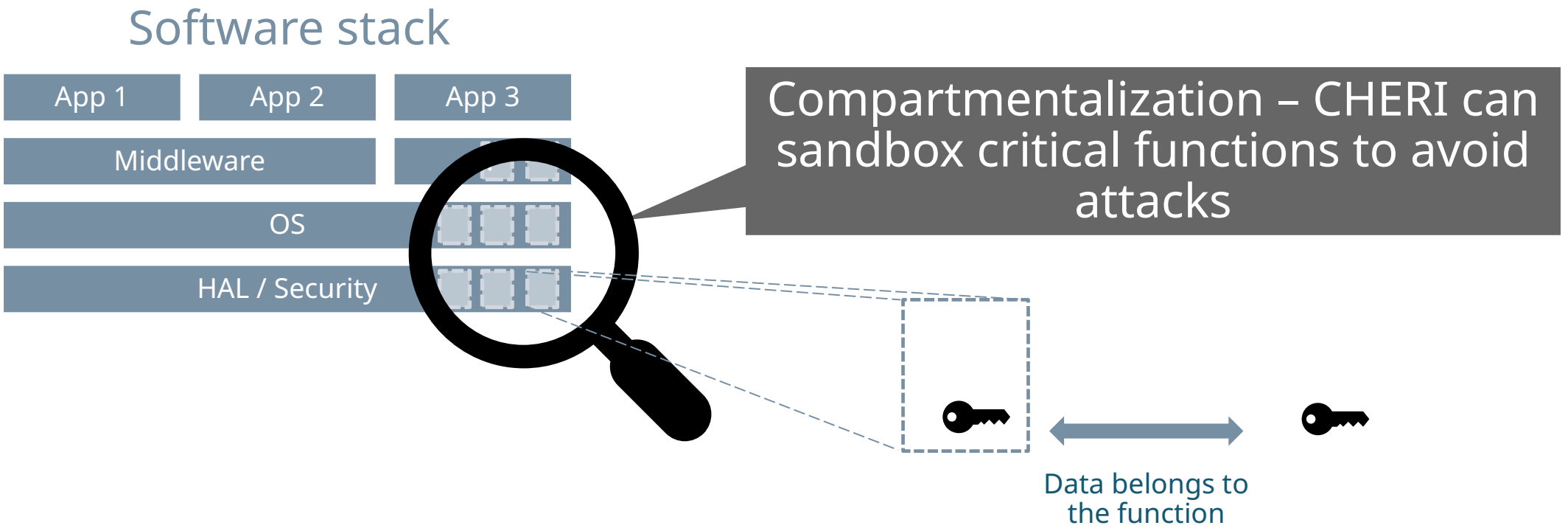
What is a capability?

- A token with rights that is used to replace a pointer
- A new architectural type



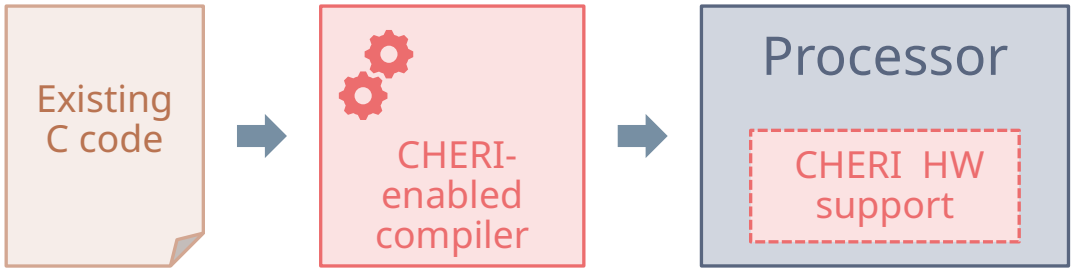
Compartmentalization

- Capabilities belong to an identified function / execution context



CHERI relies on hardware protection

- Requires adapted processor
  - Can be applied to any types of core
- Reuse existing code
  - Just recompile application
  - Choose which part to protect
- Benefit from CHERI
  - Rejects dangerous code
  - Create CHERI compartments for critical code



Adoption impact

Adoption cost

- Need new hardware
- Software effort
  - Recompile
  - Adapt very low-level code
  - Optimize security mechanisms
  - Fix issues!

Product-level

Cost	Processor ~ 4% larger
Power consumption	Similar or improved
Performance	Similar or improved
Security	Fix memory safety issues

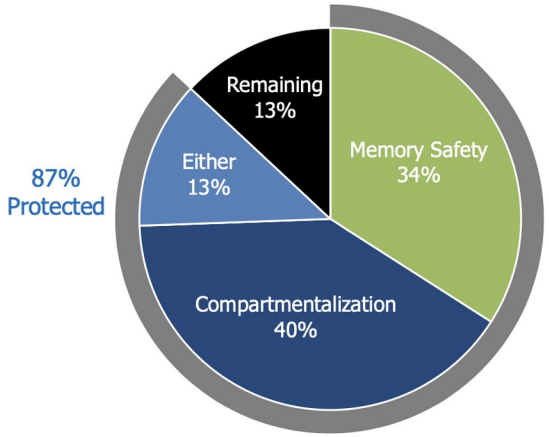
Memory safety becomes a key topic

Highlights CHERI as a solution

DARPA

Memory safety and compartmentalization would stop most campaigns

Mitigation analysis of Linux kernel high severity CVEs



- Either: if memory safety or compartmentalization were present, the bug would not be exploitable
- Remaining: memory safety or compartmentalization would not mitigate the CVE

CVE: Common Vulnerabilities and Exposures

McKee, Derrick, et al. "Preventing kernel hacks with HAKC." Proceedings 2022 Network and Distributed System Security Symposium. NDSS. Vol. 22. 2022.

Distribution Statement A: Approved for public release. Distribution is unlimited. 5

FEBRUARY 26, 2024

Press Release: Future Software Should Be Memory Safe

ONCD › BRIEFING ROOM › PRESS RELEASE

Leaders in Industry Support White House Call to Address Root Cause of Many of the Worst Cyber Attacks

CHERI projects

- A number of prototypes / proof of concept have been released
  - Proof of concept

    arm Morello Program

  - Open-source / prototype

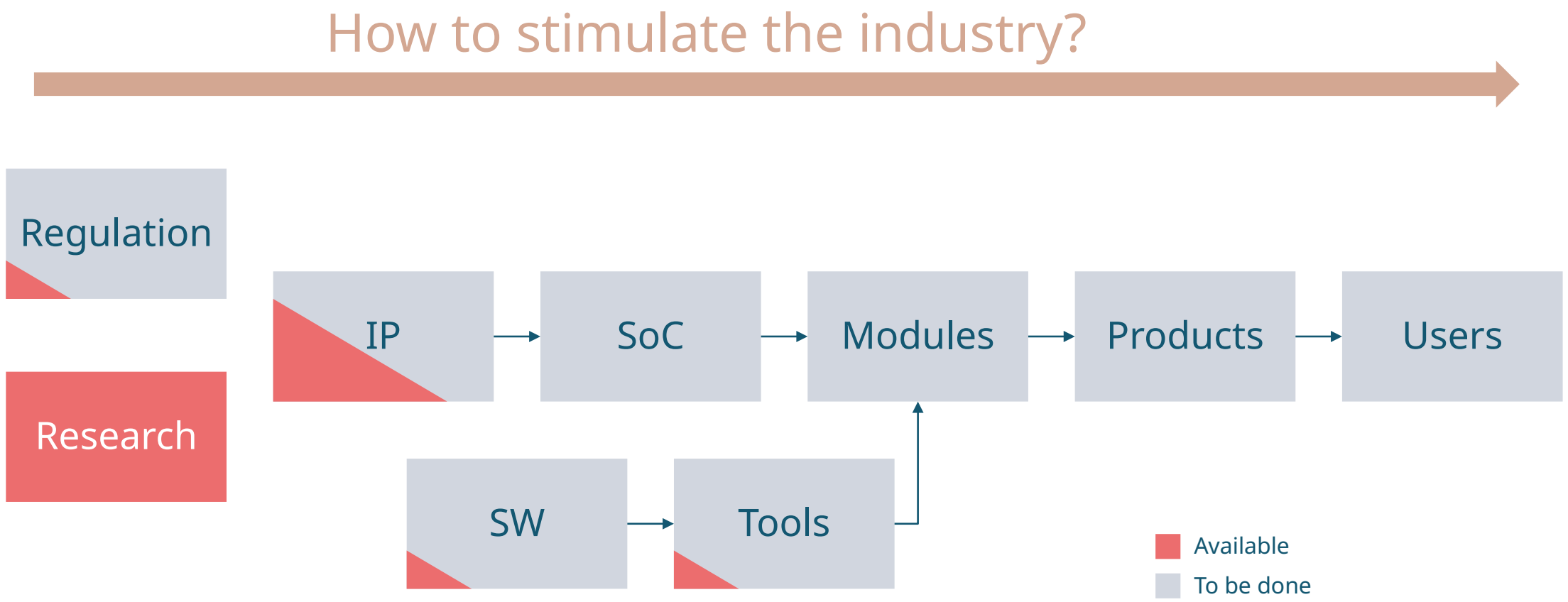
    UNIVERSITY OF CAMBRIDGE Microsoft lowRISC

  - Commercial

    Codasip

- Some OS have been ported to CHERI (Free RTOS, FreeBSD, Linux kernel…)

What is missing to get CHERI adopted?



The CHERI Alliance

Role of the CHERI Alliance

Technical alignment	Interoperability cross ISA	Compliance	Best practices
Promotion	Technical & marketing information	Media	Regulation
Community support	Open-source software support	Platform for collaboration	Education, networking

An independent entity

- ISA-agnostic
  - CHERI could be added to Arm, Intel, MIPS, RISC-V, …
- Country-agnostic
  - Initially driven by UK/US, but security is not country-dependent
- Not company-dependent
  - Represent a community
  - Open

➡ Formed as a **Community Interest Company** (CIC) *

* https://www.gov.uk/government/publications/community-interest-companies-introduction

Hosting structure

- CHERI Alliance Community Interest Company
  - Limited company – no shares
  - Defined / audited purpose
- Founding Directors of the CIC

Mike Eftimakis
VP Strategy & Ecosystem
Codasip

Robert Watson
Professor
University of Cambridge

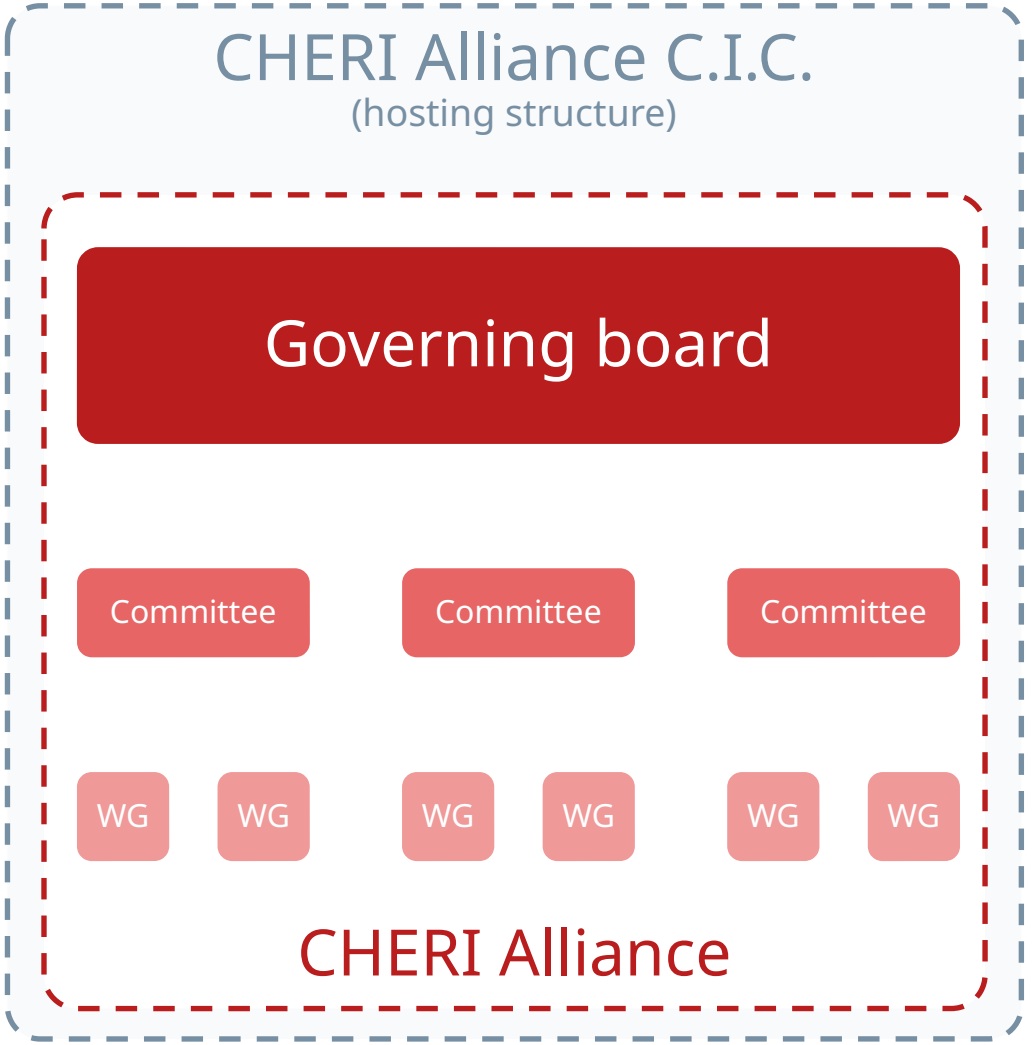
Mike Halsall
Founder
Emerging Tech Radar

Gavin Ferris
Chief Executive Officer
lowRISC

Governance

Members
(yearly membership fee)

Community
(free)



Founding Members of the Alliance

CHEVIN TECHNOLOGY

Codasip

Critical technologies inc.
Trustworthy Networked Autonomy

CYNAM

[dstl]

FreeBSD FOUNDATION

Google

LMT 光濟科技股份有限公司
LIGHT MOMENTUM TECHNOLOGY CORP.
~ Light Your Life ! ~

lowRISC

National Cyber Security Centre
a part of GCHQ

GROUP OPENHW®
PROVEN PROCESSOR IP

PARVAT

SCI

SRI

Prifysgol Abertawe
Swansea University

TRUSTED COMPUTING
CENTER OF EXCELLENCE

TechWorks

UNIVERSITY OF BIRMINGHAM

UNIVERSITY OF CAMBRIDGE

University of Glasgow

Benefits for members

Demonstrate security leadership

Network, collaborate and exchange

Accelerate adoption of CHERI

Share promotion costs

Activate the community

Conclusion

- Memory safety issues can be solved. Preventively.
- CHERI is the best solution
- CHERI is mature and needs industry adoption
- CHERI Alliance is a platform to channel this effort

Contact us!

CHERI

THANK YOU

Contact contact@cheri-alliance.net

Web www.cheri-alliance.org