Daniel S. Fowler
Secure Cyber Systems Research Group, WMG

The RESAuto Project

Investigating Capability Enhanced Microprocessors for Cyber Resilient Automotive Systems

11th July 2024

Research with Impact Education for Life

Search “WMG Warwick”

- WMG is an applied research and education faculty of the University of Warwick
- Our focus is on industry and business impact through R&D and skills provision
- We have relationships with over 1000 companies
- We provide degree education at Apprentice, UG, PG and research levels
- We run short courses and work-based learning
- Our engineering disciplines, including **resilient and smart manufacturing; sustainable materials; energy transition; transportation; data, connectivity and immersive tools**.

QR code for WMG Home Page
https://warwick.ac.uk/fac/sci/wmg/

WMG
THE UNIVERSITY OF WARWICK

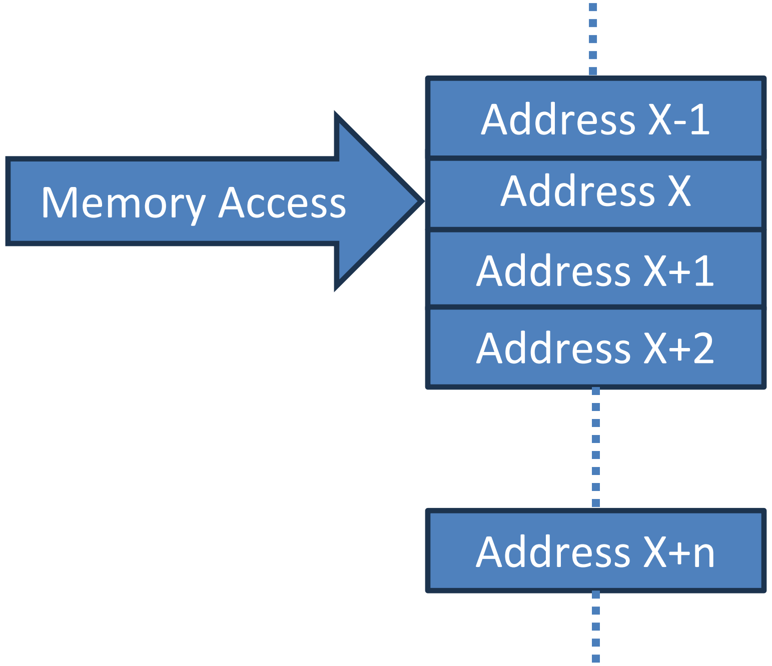
RESAuto Project - Software Resilience in Automotive Systems

- Cyber resilience for Cyber-Physical Systems (CPS).
- A CPS cannot rely on traditional enterprise-style cybersecurity of passive monitoring and reactive defence.
- A CPS must operate safely under cyber attack and demonstrate cyber resilience
- A CPS must resist attacks and degrade safely if functionality is compromised.
- For RESAuto – can **Capability Hardware Enhanced RISC Instructions** (CHERI) microcontrollers aid software resilience?

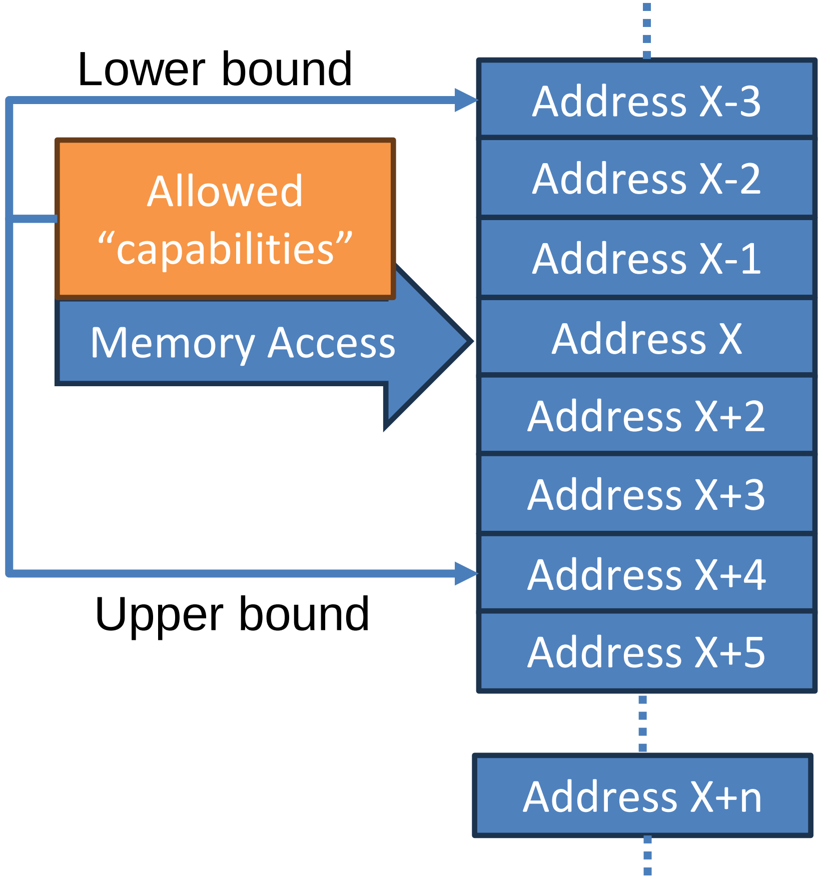
AI-generated image

What is a capability-enhanced microprocessors?

Manipulating memory access is at the root of many security issues



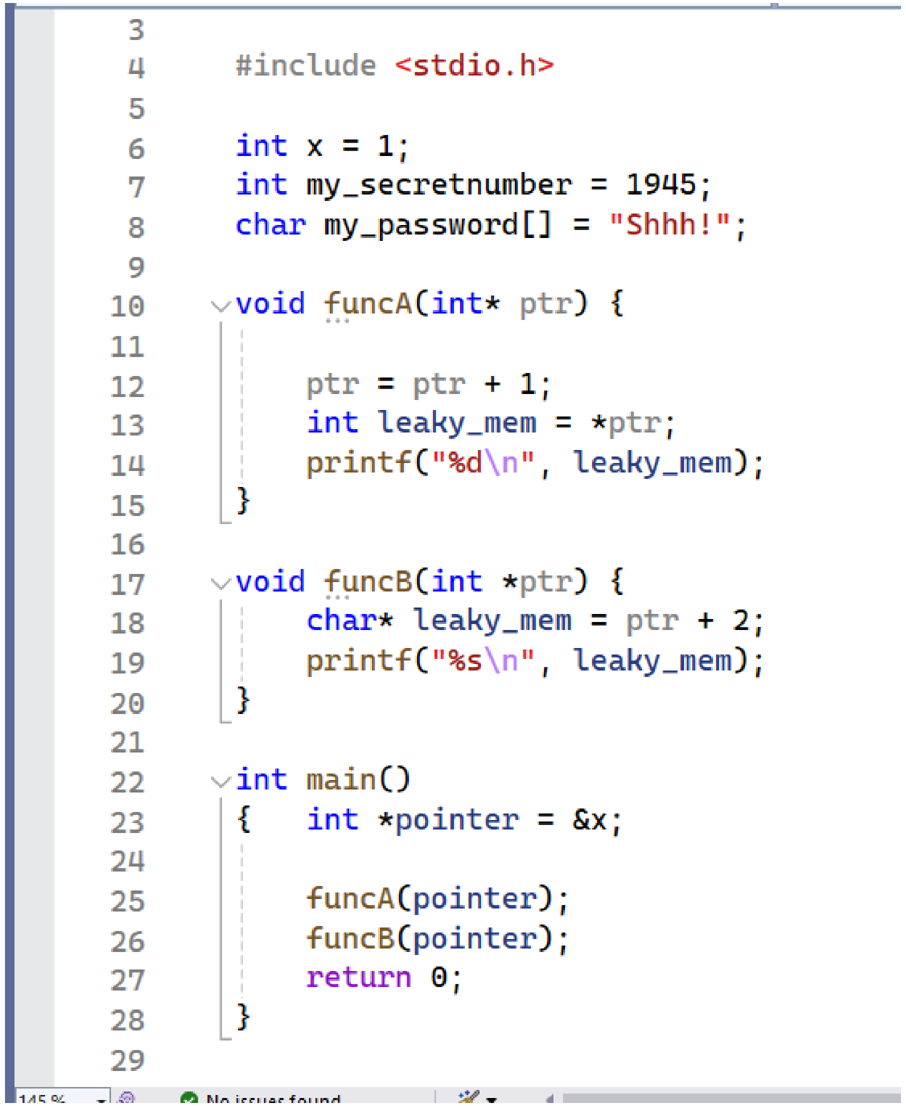
Traditional Microprocessor



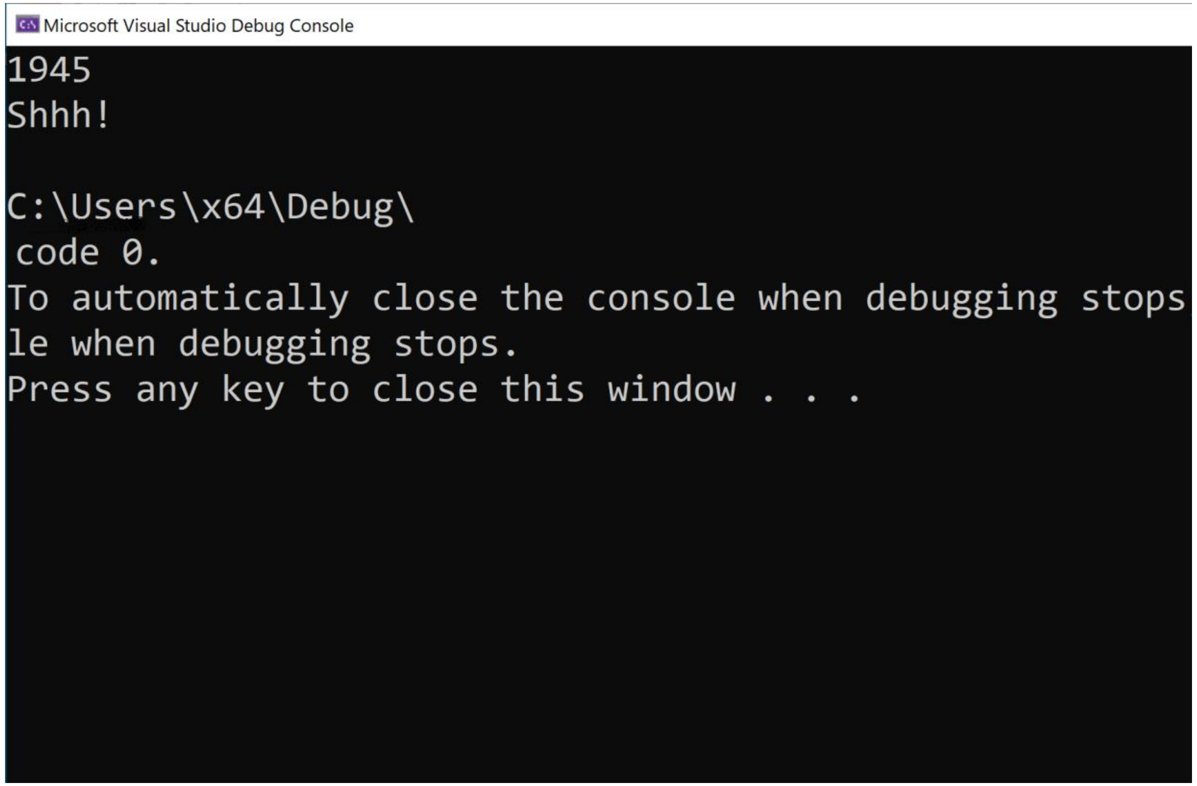
Capability Enhanced Microprocessor

Instruction tags add extra “capabilities” to memory “pointers”

“with great power there must also come – great responsibility” (Stan Lee)



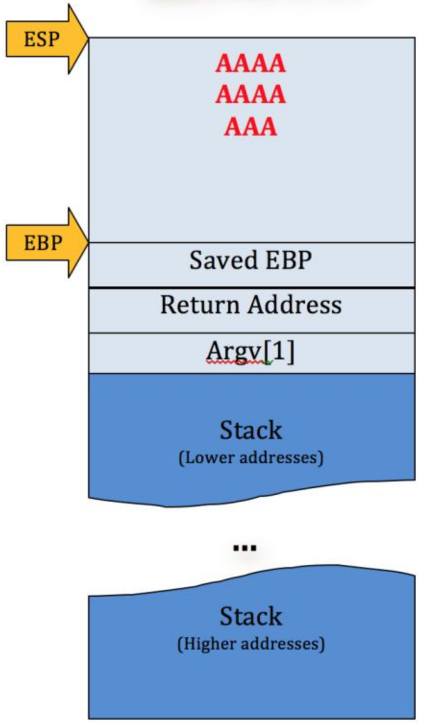
```c
#include <stdio.h>

int x = 1;
int my_secretnumber = 1945;
char my_password[] = "Shhh!";

void funcA(int* ptr) {

    ptr = ptr + 1;
    int leaky_mem = *ptr;
    printf("%d\n", leaky_mem);
}

void funcB(int *ptr) {
    char* leaky_mem = ptr + 2;
    printf("%s\n", leaky_mem);
}

int main()
{   int *pointer = &x;

    funcA(pointer);
    funcB(pointer);
    return 0;
}
```



```
1945
Shhh!

C:\Users\x64\Debug\
 code 0.
To automatically close the console when debugging stops
le when debugging stops.
Press any key to close this window . . .
```

Manipulating memory to execute malicious code

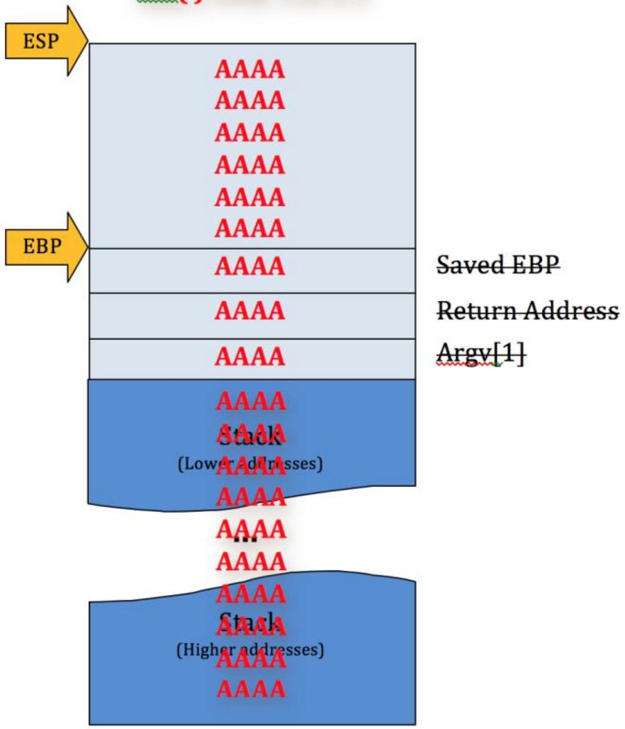
foo( ) with 11 A's



[6] CALL strcopy()

Copies value of Argv[1] into space reserved for local variable c

foo( ) with 150 A's



[6] CALL strcopy()

Copies value of Argv[1] into space reserved for local variable c

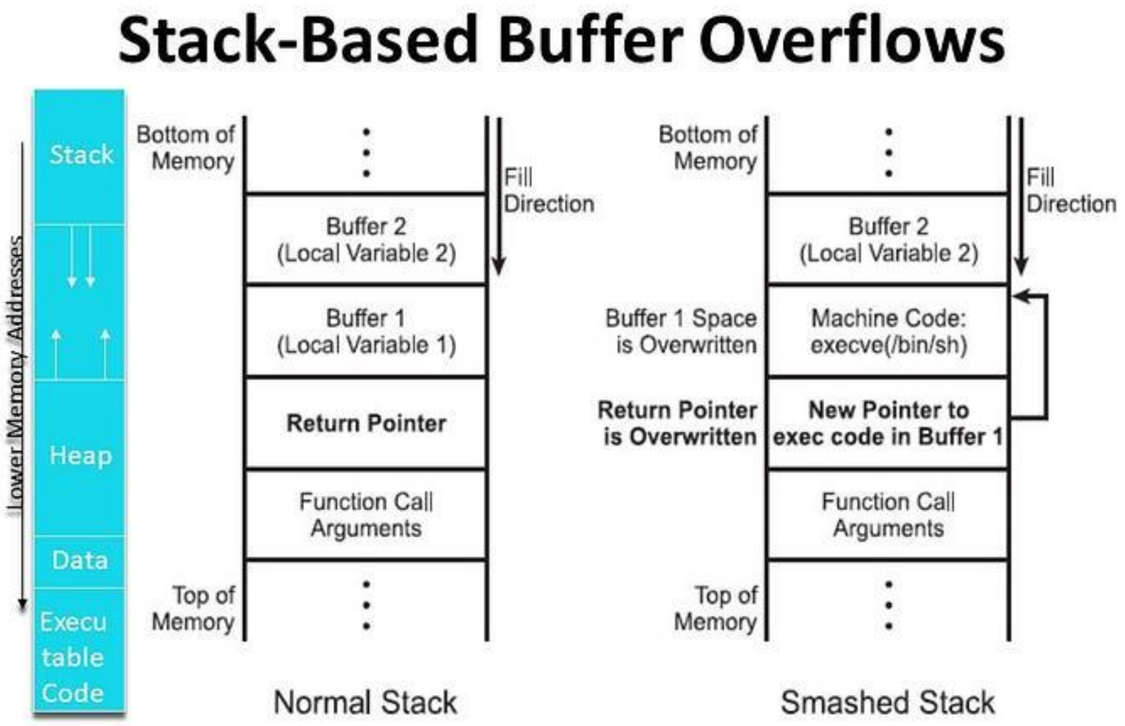
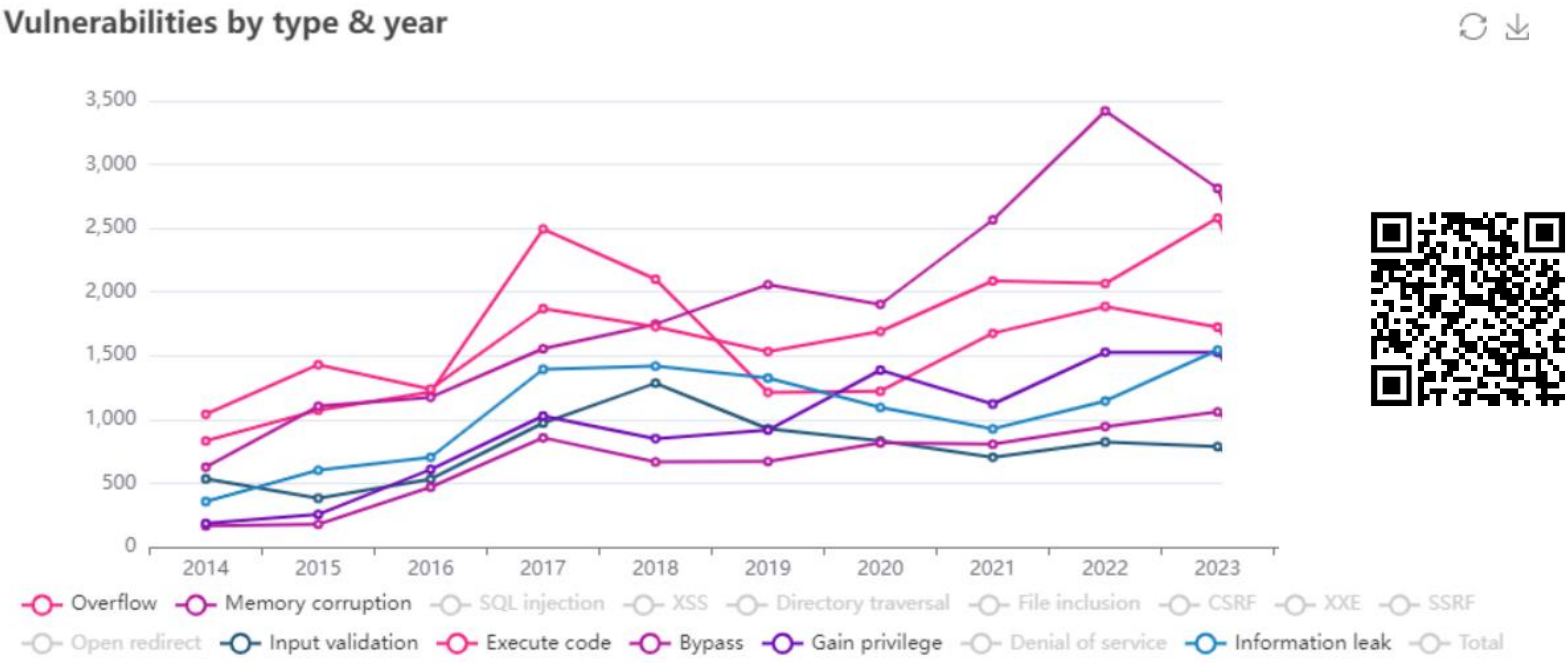

Stack-Based Buffer Overflows
Stack
Heap
Data
Executable Code
Lower Memory Addresses
Bottom of Memory
Buffer 2 (Local Variable 2)
Buffer 1 (Local Variable 1)
Return Pointer
Function Call Arguments
Top of Memory
Fill Direction
Normal Stack
Bottom of Memory
Buffer 2 (Local Variable 2)
Buffer 1 Space is Overwritten
Machine Code: execve(/bin/sh)
Return Pointer is Overwritten
New Pointer to exec code in Buffer 1
Function Call Arguments
Top of Memory
Fill Direction
Smashed Stack

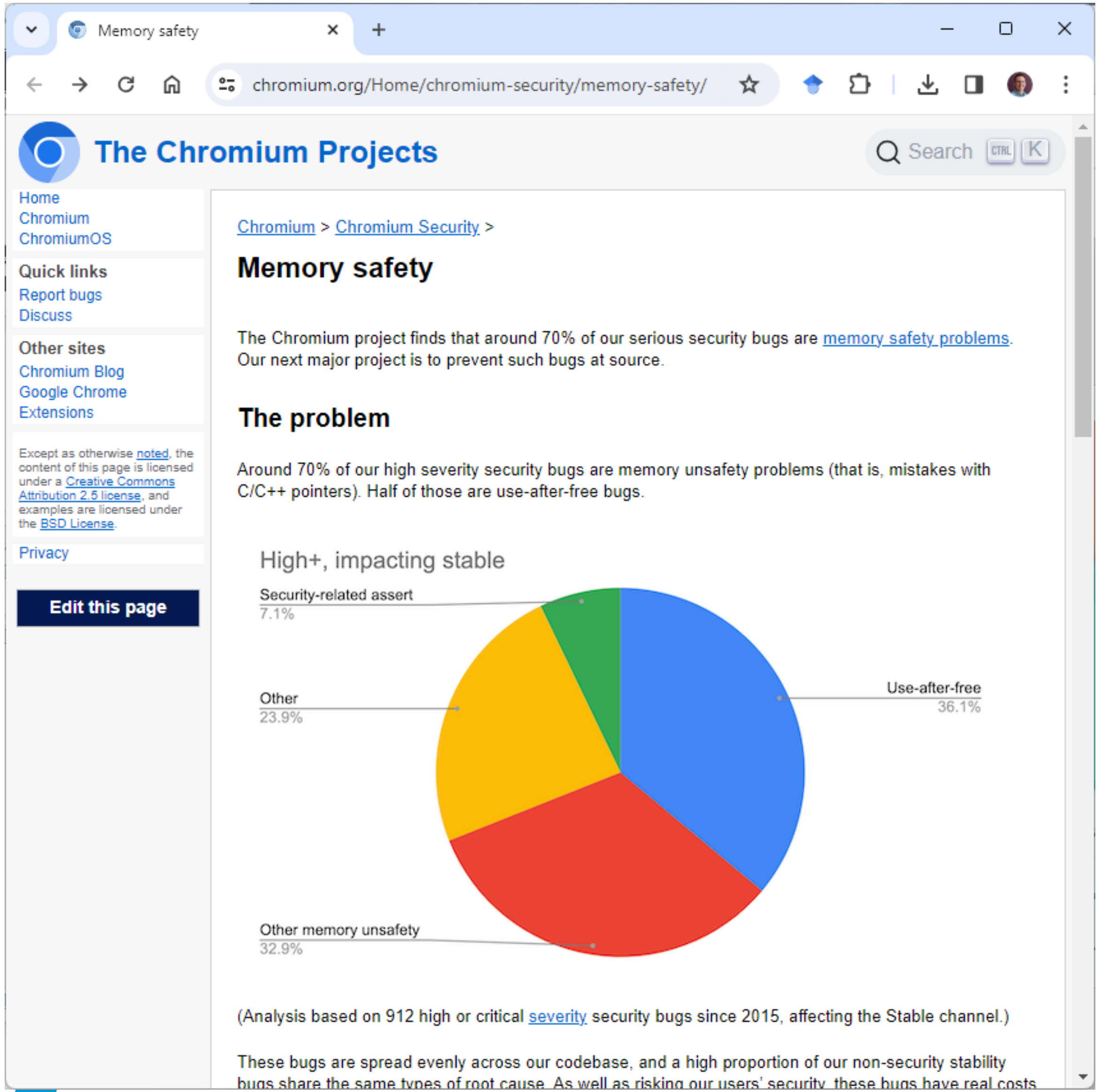
EXHIBIT 10.2 A normal stack and a stack with a buffer overflow.

Common Vulnerabilities and Exposures (CVE)



https://www.cvedetails.com/vulnerabilities-by-types.php

Acknowledging the memory bug issue


Memory safety
chromium.org/Home/chromium-security/memory-safety/
The Chromium Projects
Search CTRL K
Home
Chromium
ChromiumOS
Quick links
Report bugs
Discuss
Other sites
Chromium Blog
Google Chrome Extensions
Except as otherwise noted, the content of this page is licensed under a Creative Commons Attribution 2.5 license, and examples are licensed under the BSD License.
Privacy
Edit this page

Chromium > Chromium Security >

Memory safety

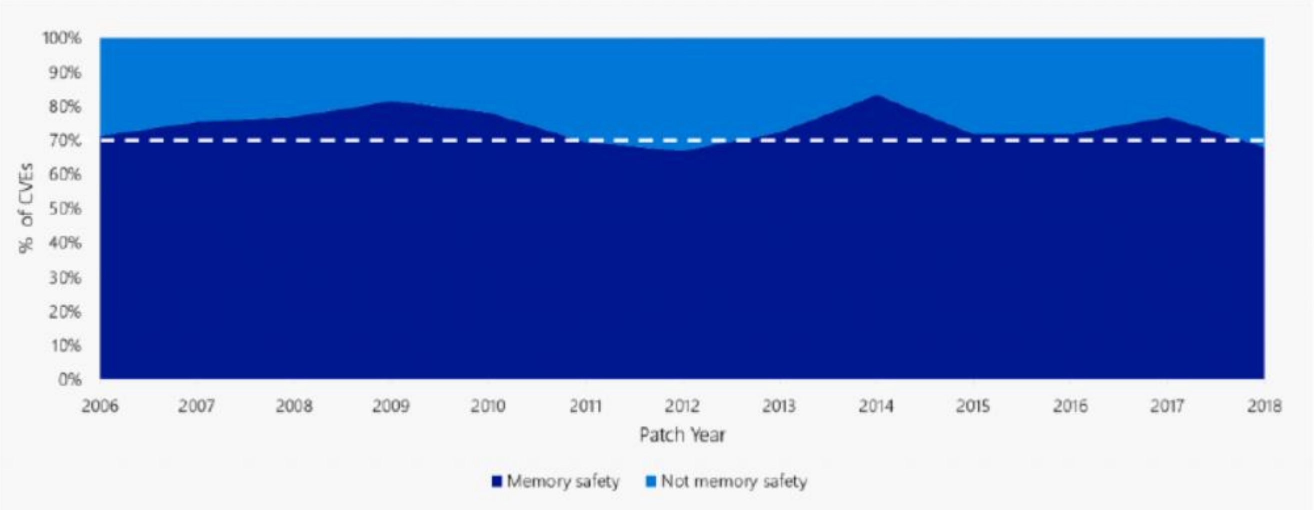
The Chromium project finds that around 70% of our serious security bugs are memory safety problems. Our next major project is to prevent such bugs at source.

The problem

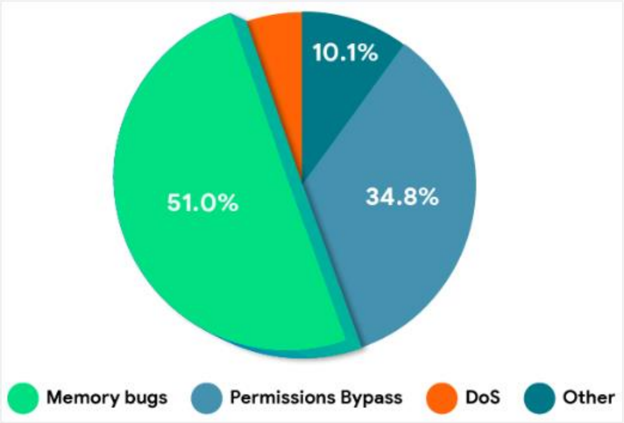
Around 70% of our high severity security bugs are memory unsafety problems (that is, mistakes with C/C++ pointers). Half of those are use-after-free bugs.

(Analysis based on 912 high or critical severity security bugs since 2015, affecting the Stable channel.)

These bugs are spread evenly across our codebase, and a high proportion of our non-security stability bugs share the same types of root cause. As well as risking our users' security, these bugs have real costs



~70% of the vulnerabilities Microsoft assigns a CVE each year continue to be memory safety issues



Memory safety bugs
contribution to Android vulnerabilities

Common Vulnerabilities and Exposures (CVE)

A Few Memory Related CVEs Associated with Vehicles

CVE-2023-39076 - Injecting data into the USB memory area causes a Denial of Service (DoS) in the in-car infotainment system - vehicles from GM Chevrolet

CVE-2023-32157 - Heap-based Buffer Overflow Arbitrary Code Execution Vulnerability, lack of proper validation of the length of user-supplied data prior to copying it to a fixed-length heap-based buffer - vehicles from Tesla.

CVE-2023-32155 - Write past the end of an allocated buffer results in Out-Of-Bounds Write Local Privilege Escalation - vehicles from Tesla

CVE-2023-28885 - Denial of service via temporary failure of Media Player with a crafted MP3 file - vehicles from GM Chevrolet

CVE-2021-23910 - There is an out-of-bounds array access in RemoteDiagnosisApp - veicles from Mercedes-Benz

CVE-2021-23906 - A Message Length is not checked in the HiQnet Protocol, leading to remote code execution - vehicles from Mercedes-Benz

CVE-2020-27524 - Memory content leaks - vehicles from Audi

CVE-2020-16142 - Bluetooth stack mishandles %x and %c format-string specifiers - vehicle from Mercedes-Benz

CVE-2019-13582 - A stack overflow could lead to denial of service or arbitrary code execution - vehicles from Tesla

CVE-2019-13581 -A heap-based buffer overflow allows remote attackers to cause a denial of service or execute arbitrary code via malformed Wi-Fi packets - vehicles from Tesla

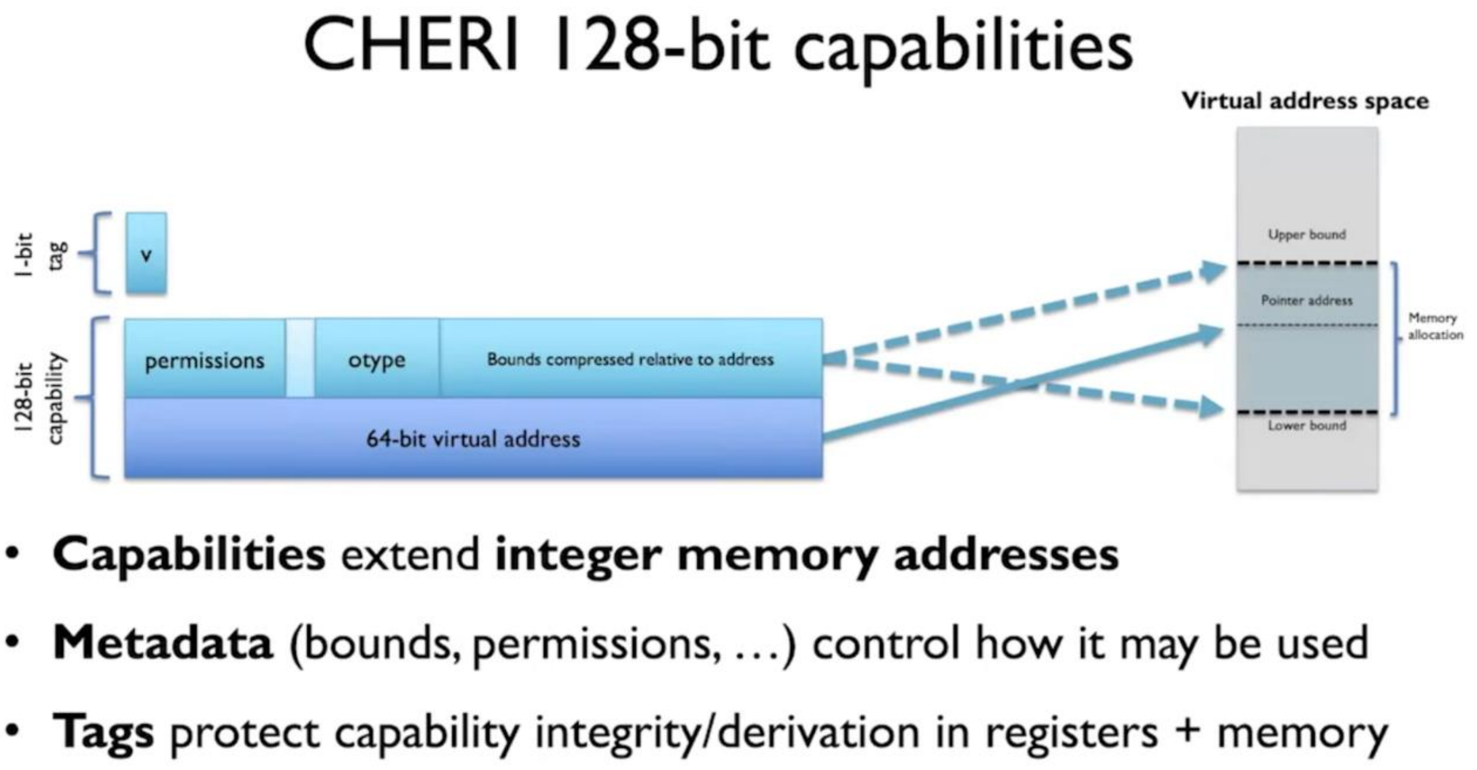
CVE-2017-9647 - A Stack-Based Buffer Overflow issue was discovered - vehicles from BMW, Ford, Infiniti, and Nissan

CVE-2017-9633 - An Improper Restriction of Operations within the Bounds of a Memory Buffer issue was discovered - vehicles from BMW, Ford, Infiniti, and Nissan

CVE-2012-2619 - Out-of-bounds read - vehicles from Ford

Ref: https://cve.mitre.org/cve/search_cve_list.html

CHERI – New capabilities for memory access

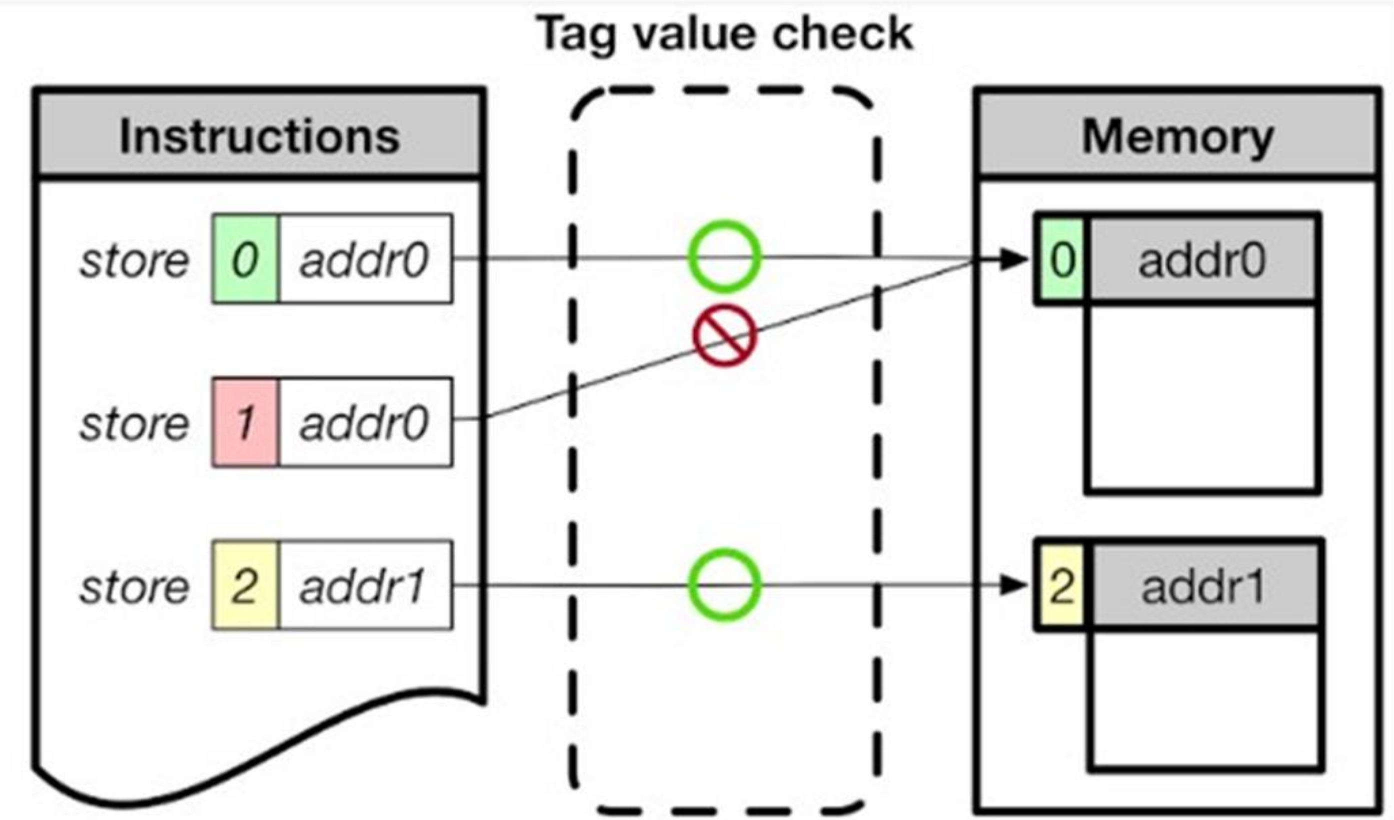


CHERI 128-bit capabilities

- **Capabilities** extend **integer memory addresses**
- **Metadata** (bounds, permissions, …) control how it may be used
- **Tags** protect capability integrity/derivation in registers + memory

CHERI pointers – 2x the size of traditional software pointers
e.g., 128 bits on a 64-bit system + a validity tag bit

Another example CHERI capability - memory tagging



CHERI, Capability Hardware Enhanced RISC Instructions

A Capability Design

- E.g., the “CHERI” design from SRI International and the University of Cambridge.
- Implemented by the Arm “Morello” program.
- Morello is a prototype system-on-chip (SoC) and a development board that adopts the CHERI design.
- Are such “capability-enhanced” processors suitable for embedded device microcontrollers? E.g., vehicle electronic control units (ECUs).

UNIVERSITY OF CAMBRIDGE

arm

https://www.cl.cam.ac.uk/research/security/ctsrd/cheri/

ResAuto Partners

THALES
Building a future we can all trust

Lead Partner

WMG
THE UNIVERSITY OF WARWICK

Demonstrator

TechWorks
AESIN
AUTOMOTIVE ELECTRONICS INNOVATION

Dissemination

Funding: UKRI Innovate UK

Examining economic and business implications

- New tooling – compilers and debuggers – Commodity compilers are available

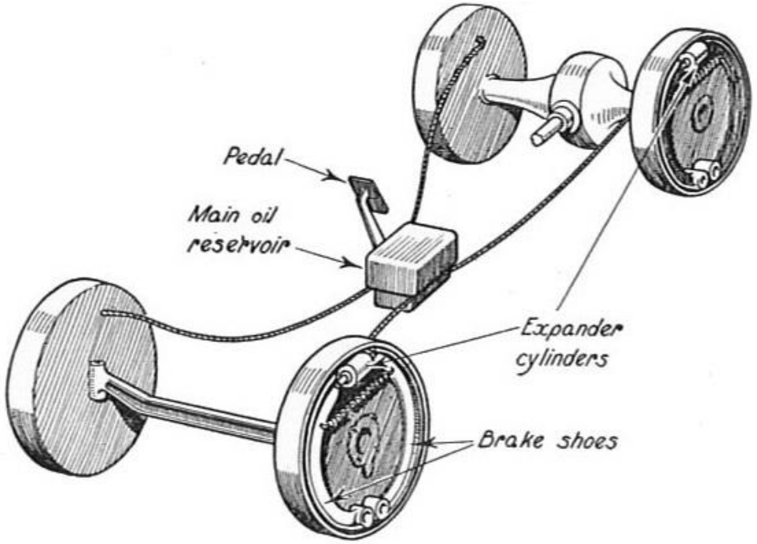
- Requires additional programming knowledge – training considerations for future developers

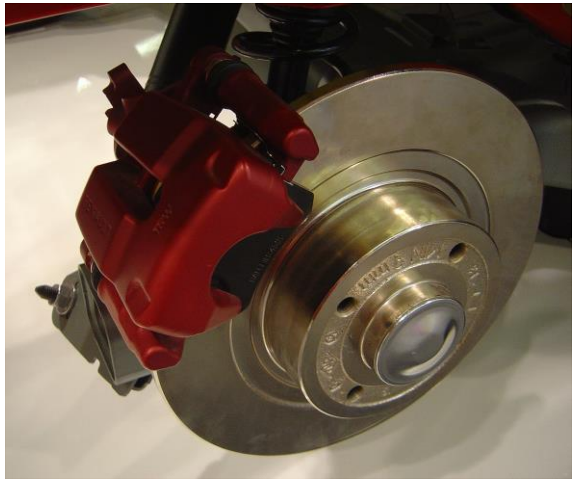
- What is the impact on the supply chain?
  1. Intellectual Property (IP) Providers – compilers, software, libraries, tools, testing, equipment, secure coding guidelines
  2. Semiconductor/chip companies and tier suppliers – electronics, modules, and parts
  3. Original Equipment Manufacturers (OEMs) – the vehicle manufacturers

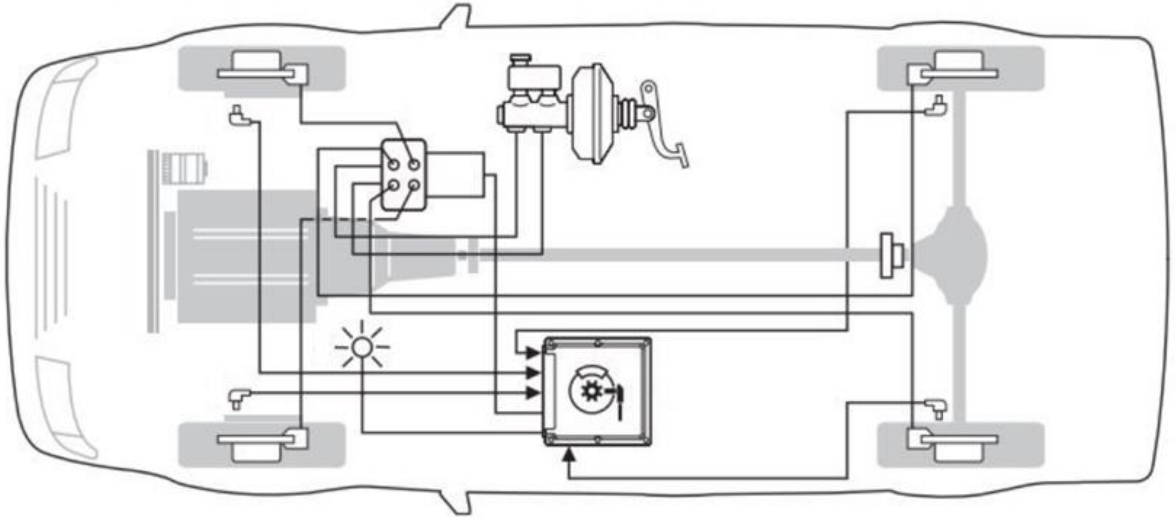
- The structural changes required to the engineering processes for these three groups

- What are the ethical, insurance and legal issues if organisations rely too much on the chip hardware for security?

Increasing system complexity - automotive braking



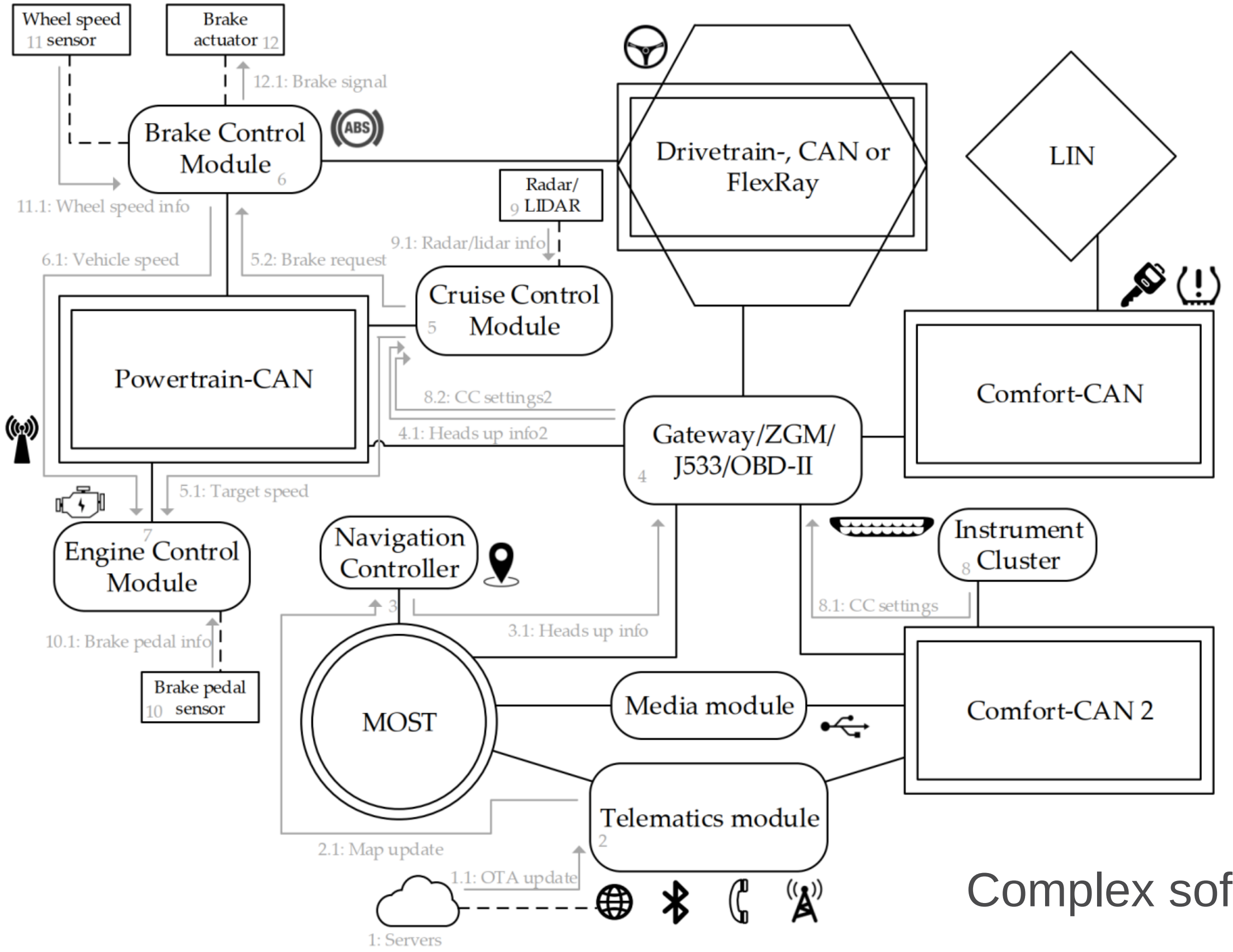




Refs:
cart-brake.jpg, Public Domain, Wikimedia Commons, Holz-Bremse
disc_brake.jpg, CC-BY-SA 3.0, Wikimedia Commons, Disk brake
hydraulic_brake_system.jpg, Public Domain, Wikimedia Commons, Lockheed hydraulic brake system
ecu_brake_system.jpg, Automotive Handbook, 11th Edition, Robert Bosch GmbH

WMG
THE UNIVERSITY OF WARWICK

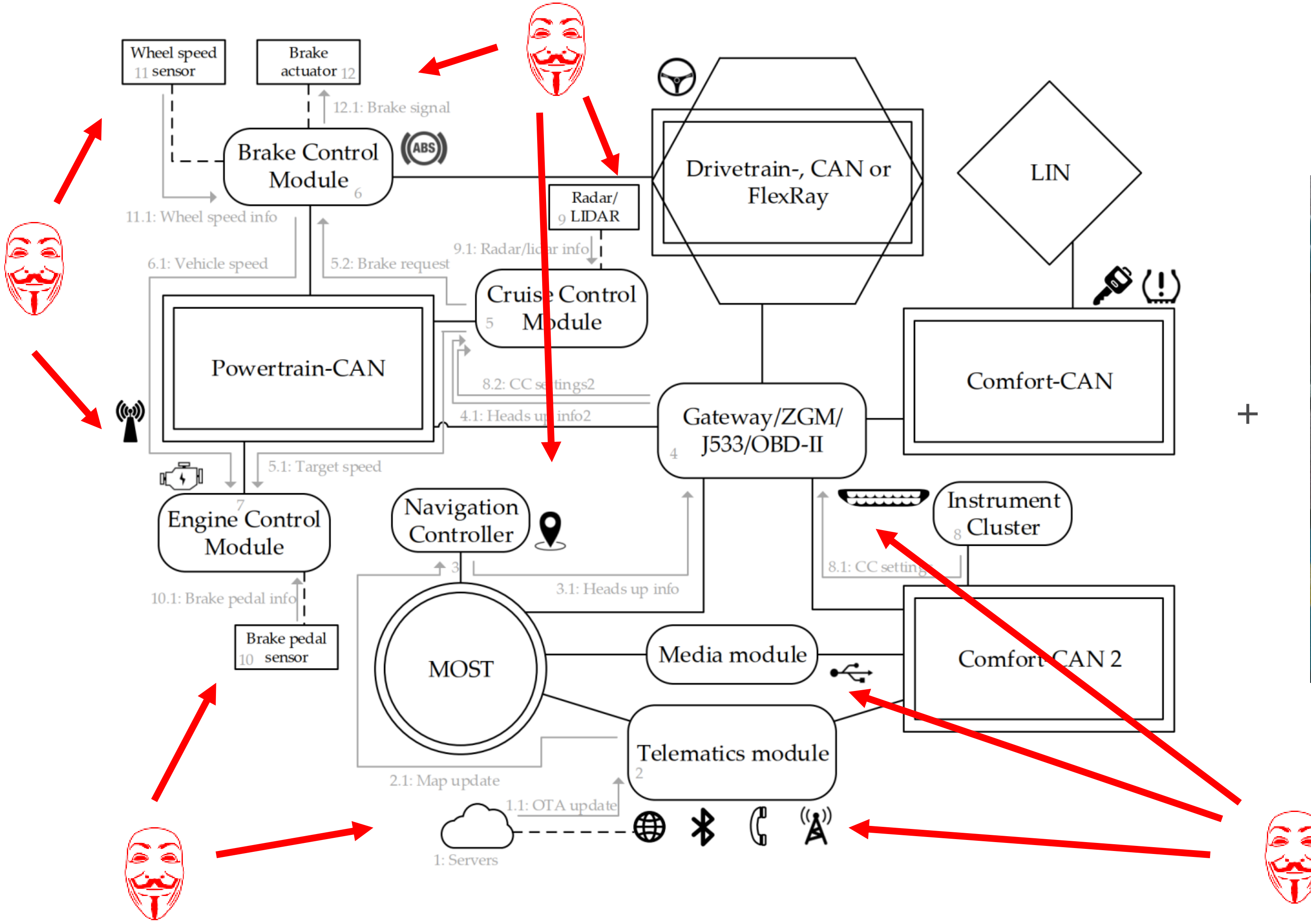
The automotive braking system example



Complex software control + Artificial Intelligence

Ref: Winsen, S., ‘Threat modelling for future vehicles: on identifying and analysing threats for future autonomous and connected vehicles’

Complexity increases the cyber attack surface

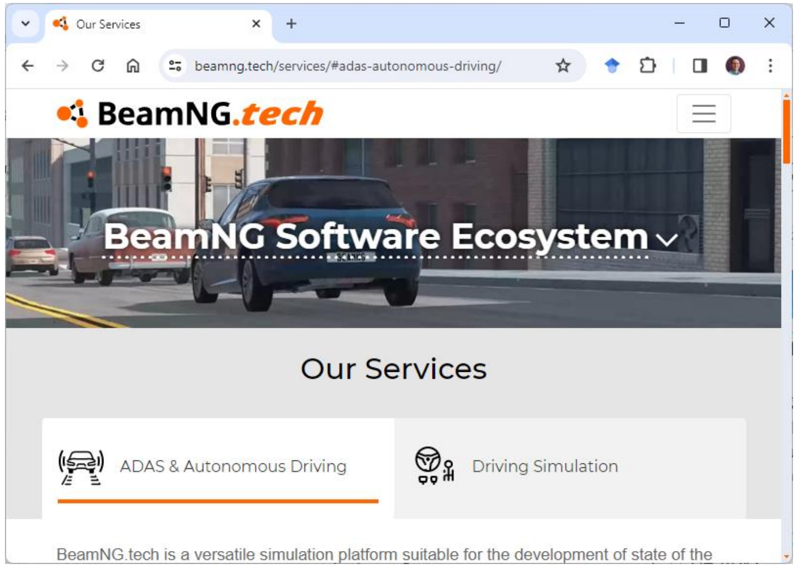


+

THE UNIVERSITY OF WARWICK

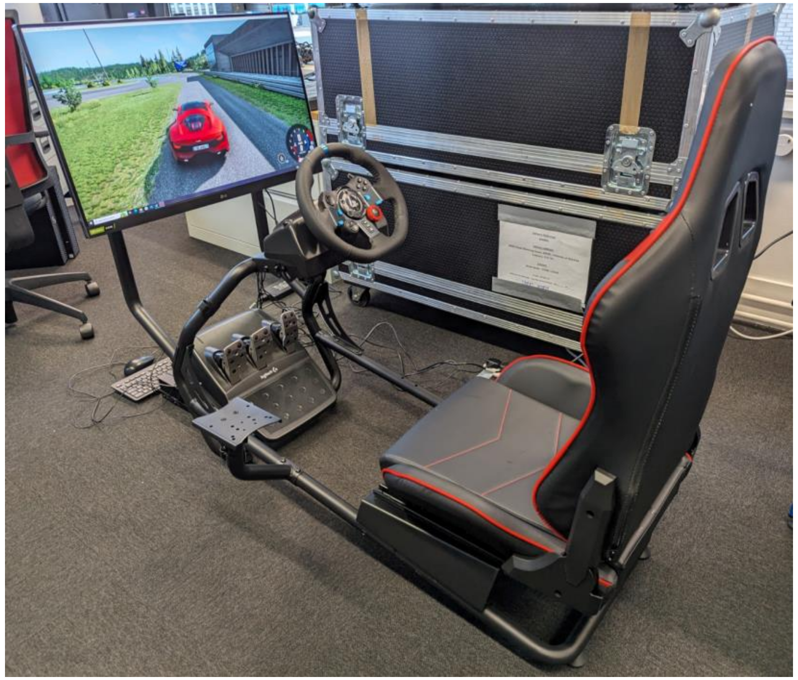
WMG CHERI Braking Demonstrator

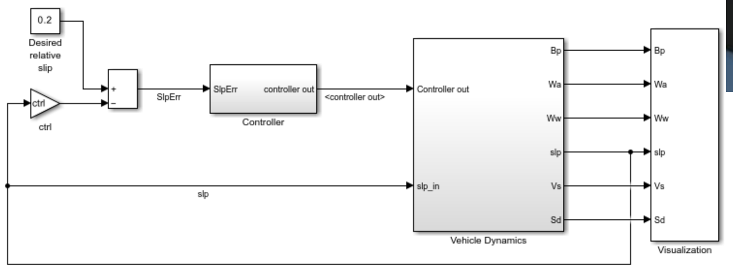
BeamNG simulator for
braking source and sink data.


Our Services
beamng.tech/services/#adas-autonomous-driving/
BeamNG.tech
BeamNG Software Ecosystem
ADAS & Autonomous Driving
Driving Simulation

Our Services

BeamNG.tech is a versatile simulation platform suitable for the development of state of the



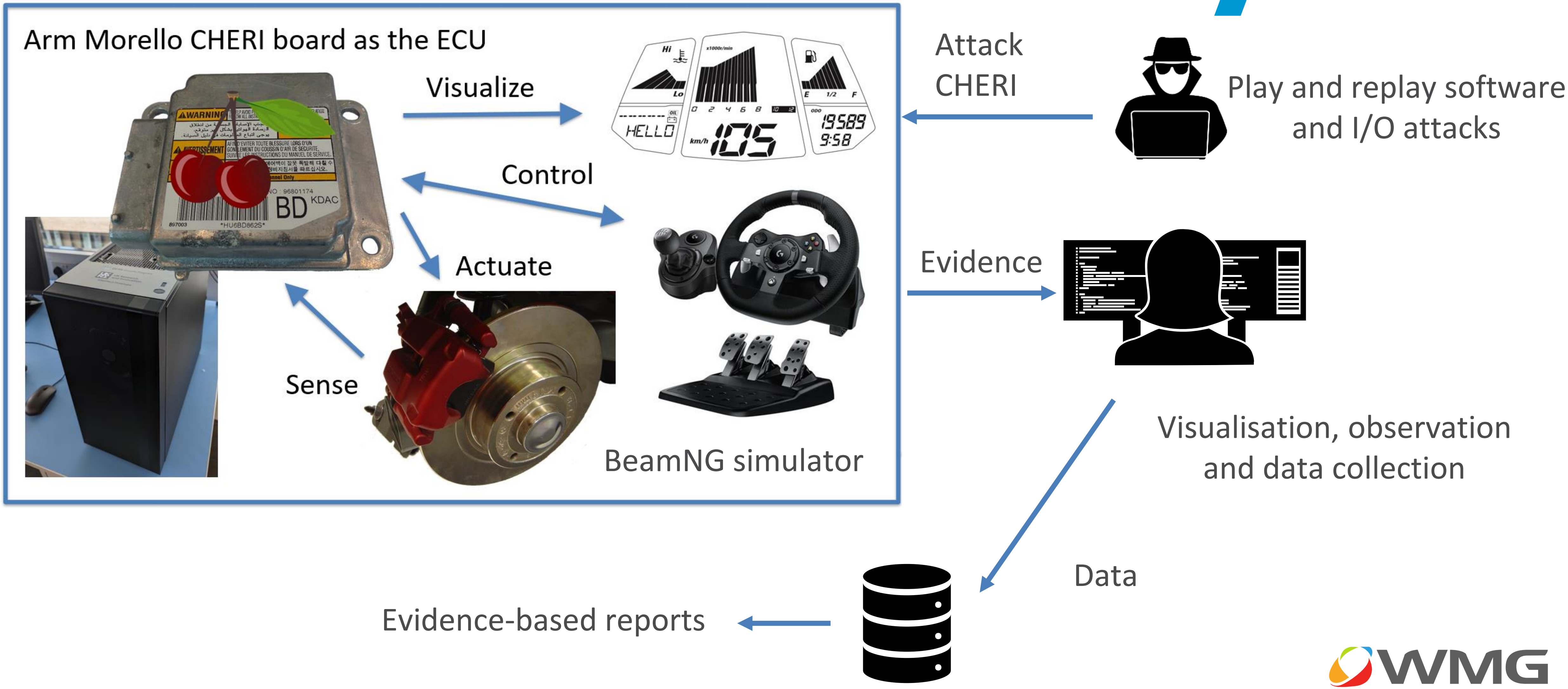


It’s here at AESIN!

ABS model in C
on CHERI “ECU”

WMG
THE UNIVERSITY OF WARWICK

Arm Morello board (CHERI) as a braking ECU demonstrator



Arm Morello CHERI board as the ECU

Visualize

Control

Actuate

Sense

BeamNG simulator

Attack CHERI

Play and replay software and I/O attacks

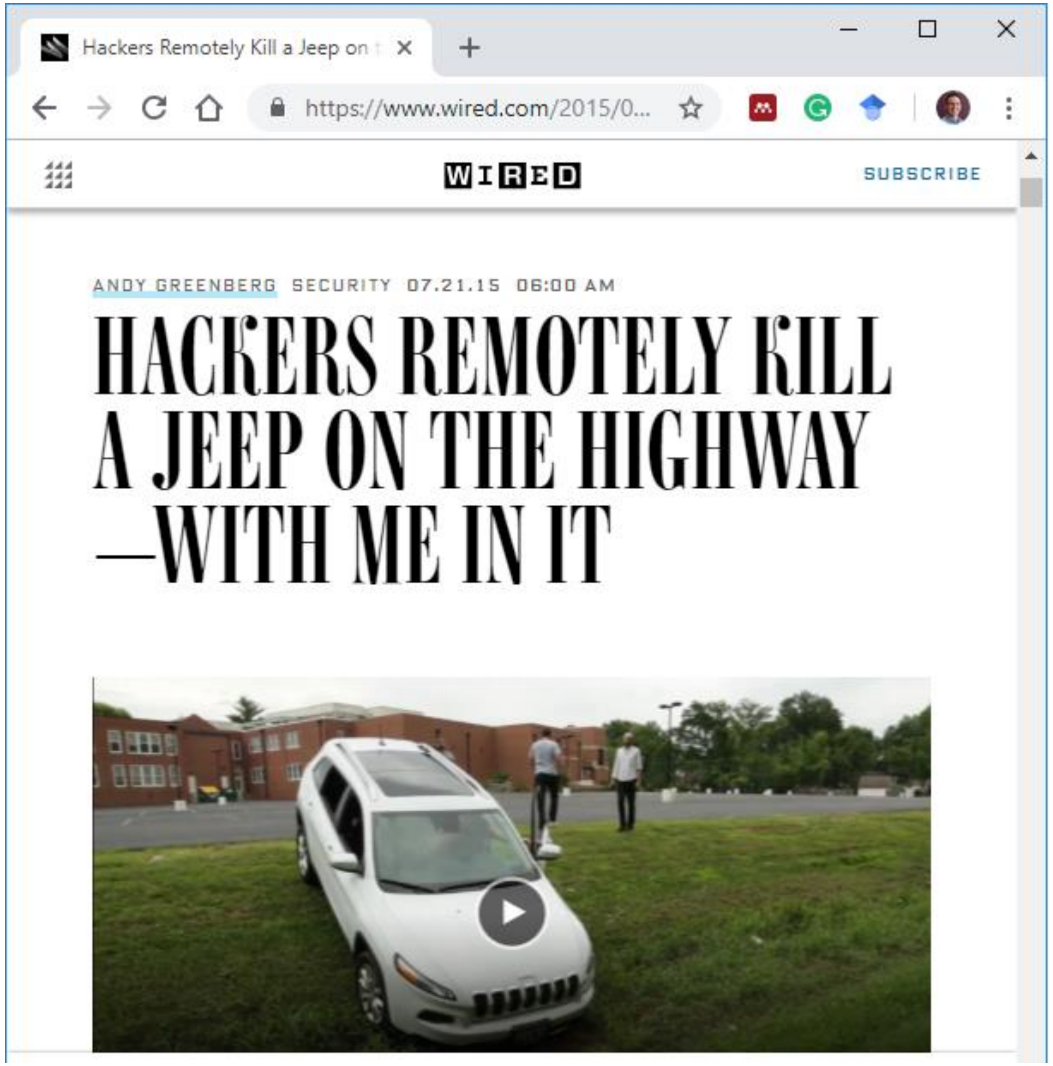
Evidence

Visualisation, observation and data collection

Data

Evidence-based reports

The Jeep hack was not a memory vulnerability



Hackers Remotely Kill a Jeep on t
https://www.wired.com/2015/0...
WIRED
SUBSCRIBE

ANDY GREENBERG SECURITY 07.21.15 06:00 AM

HACKERS REMOTELY KILL A JEEP ON THE HIGHWAY —WITH ME IN IT

Memory manipulation is not the only issue

The seminal 2015 Jeep hack was down to poor security controls and a lack of authentication checks – no memory manipulation was required

CHERI would be part of the solution

ResAuto Partners

THALES
Building a future we can all trust

Lead Partner

WMG
THE UNIVERSITY OF WARWICK

Demonstrator

TechWorks
AESIN
AUTOMOTIVE ELECTRONICS INNOVATION

Dissemination

Please gives us your thoughts:

dan.fowler@warwick.ac.uk

Funding: UKRI Innovate UK