Improving Europe's cybersecurity posture through memory safety

Supporters

- Internet Security Research Group
- Tauri
- Rust Foundation
- Special Interest Group Cybersecurity of ICT
- Research Platform Netherlands (IPN) and ACCSS
- Trifecta Tech Foundation
- Tweede Golf
- Stackable

Contributors

- Benjamin Schilling
- Christian (fukami) Horchert, CrabNebula Ltd.
- Rebecca Rumbul, Rust Foundation
- Josh Aas, Internet Security Research Group
- prof. dr. H.J. Bos, Vrije Universiteit Amsterdam
- Erik Poll, Radboud University

Authors:
Tara Tarakiyee
Hugo van de Pol

Executive Summary

The number of cybersecurity incidents that affect European citizens and businesses is rising at an alarming rate [1]. 70% of the vulnerabilities in major digital systems built on decades-old technologies share the same root cause and can be prevented by using modern, memory-safe technology [2].

This technology is mature, perfectly fits Europe's forthcoming secure-by-design approach to cybersecurity, and is the most effective way to protect Europe's cybersecurity, to reduce cybersecurity costs, and to foster innovation.

However, its adoption rate is slow due to a lack of short-term economic incentives. We've now left the door wide open: attackers eagerly exploit vulnerabilities in our major digital systems.

The supporting organisations call on European and national policymakers to act, out of obligation as well as untapped opportunity: to provide clear incentives and support for the large-scale adoption of memory-safe technology.

Introduction

Today, the authoring organisations are releasing this joint statement underscoring the critical importance of adopting memory-safe technology as a foundational pillar of digital infrastructure that is secure by design.

Enisa reports [1] that digital threats stemming from vulnerabilities are escalating at an alarming rate. Using memory-safe technology is crucial for building systems that are secure-by-design. In prior years, the U.S. Cybersecurity and Infrastructure Security Agency (CISA) and the U.S. White House have outlined this in numerous publications [e.g. 3, 4, 5].

The signatories call for European and national policy makers to acknowledge that the large-scale adoption of memory-safe tech in Europe is crucial, and to act accordingly by providing the incentives needed to substantially increase the current rate of adoption.

What Is Memory Safety?

Memory safety refers to the ability of programming languages — the very building blocks of all of the digital systems we rely on every day — to prevent security vulnerabilities that are due to accidentally mishandling memory.

These memory-related vulnerabilities:

- Account for 60%–70% of all software security vulnerabilities [2, 6] in large codebases written in memory-unsafe languages. Examples of such code bases include the operating systems, web browsers, and screen-sharing solutions that businesses and citizens use every day;
- Are typically hard to detect, costly to fix when they appear in production, and often used in exploits [7, 8, 9, 10];
- Persist despite decades of developer training and tooling aimed at preventing them.

Memory-safe languages (e.g., Rust, Python, Swift, or Java) mitigate these risks by enforcing strict rules, making it impossible to introduce entire classes of memory-related bugs by default. This makes memory safety one of the most powerful methods for achieving secure-by-design systems.

Why Memory-Safe Tech Is Crucial

Choosing memory-safe technologies is one of the most impactful decisions a manufacturer of digital infrastructure components can make to improve cybersecurity.

- Security vulnerabilities affect everyone: From online banking to airport operation, large parts of everyone's lives have been affected by security vulnerabilities already. With existing

technology, up to 70% could have been mitigated from the beginning by applying memory-safe technology.

- Security becomes scalable: It reduces reliance on human vigilance, which is increasingly unsustainable in the face of talent shortages, increasingly complex systems, and growing attack surfaces.

- Prevention becomes the default: By enforcing security at the language level, the software becomes safer by default, not by afterthought.

- Ecosystems around memory-safe languages improve developer efficiency: Developer experience of all major memory-safe languages contributes to increased productivity, as they provide integrated tools for development and supply chain management.

The Urgency for Action: Secure-by-Design

We are at a pivotal moment:

- Global tensions and cyber warfare are on the rise, while our dependence on digital infrastructure grows daily;
- The cost of cyber breaches continues to climb, with some incidents costing billions;

- There is an acute shortage of cybersecurity professionals and qualified software engineers [11].

All of this demands a shift in mindset: from reactive patching to proactive prevention [12].

Considering security earlier in the development lifecycle is not a new concept, but it is certainly no longer optional. This has been recognised in Europe by the passing of the Cyber Resilience Act (CRA) and its security-by-design requirement.

Where We Are Today

As CISA previously concluded, the most effective, systematic, and scalable way to get rid of all memory-safety vulnerabilities is to build new systems using memory-safe by default tech, and to migrate legacy systems step by step.

In Europe, there is now some adoption of modern memory-safe tech and several incentives for it, including:

- Regulation that demands a secure-by-design approach to cybersecurity (e.g., the CRA, which is in development);
- The awareness that the current reactive approach to cybersecurity is no longer feasible [12];
- The generally high skill level of engineers, which lowers the bar for using modern tech.

At the same time, individual businesses, including large industry players that build part of our digital infrastructure, perceive a lack of short-term economic incentive in the midst of fierce international competition to start migrating parts of large existing code bases. costs of cybersecurity, both for companies and for society, and ensuring technological innovation, thus strengthening Europe's competitive position.
Society as a whole is bearing the long-term consequences and costs of the persisting vulnerabilities in these systems. In the Netherlands for example, the root cause of both the recent hack of the Public Prosecution office and a foreign state actor spying on the Ministry of Defense in 2022-2023 was memory-unsafe code.

To us this means that regulatory bodies have the obligation to recognize the detriment of the current situation, as well as the opportunities that faster adoption of memory-safe technologies would bring: lowering the costs of cybersecurity, both for companies and for society, and ensuring technological innovation, thus strengthening Europe's competitive position.

We Call On:

- **Decision-makers to take a decisive stance:** Formally recognise memory safety as a cornerstone of Europe's cyber resilience strategy and include specific memory safety requirements in the implementation guidelines for the Cyber Resilience Act. Where existing memory-unsafe languages are in use, this would include recommendations on measures to reduce risk where feasible.

- **Industries, to understand the economic and security business case for memory safety:** Look into the long-term financial advantages of memory-safe technologies, including reduced security incident costs, decreased downtime, lower maintenance burdens, and enhanced customer trust.

- **Standardisation Bodies, to establish clear migration pathways:** Issue guidelines and strategies to help organisations assess and transition critical digital infrastructure components to memory-safe technologies that build on best practices and existing work in open standards bodies.

- **EU and Governments, to invest in capacity building, provide transition support and require risk assessment and mitigation:** Create and fund specialised training programs to address the skill shortage in memory-safe programming, partnering with universities to align outputs

with the future needs of industry. Establish funding mechanisms and technical resources to assist small and medium enterprises in evaluating and implementing memory-safe alternatives in their development processes. Invest in existing and potential open-source projects and companies developing memory-safe technologies through targeted support and public-private partnerships. Make memory safety evaluations and mitigation planning a mandatory component of security risk assessments for critical infrastructure providers and government suppliers.

Other References

2025 Open Source Security and Risk Analysis Report:
https://www.blackduck.com/resources/analyst-reports/open-source-security-risk-analysis.html

The Time is Now - Practical Mem-Safety:
https://github.com/dwizzzle/Presentations/blob/master/david_weston-isrg_tectonics_keynote.pdf

Securing tomorrow's software: the need for memory safety standards:
https://security.googleblog.com/2025/02/securing-tomorrows-software-need-for.html

The Memory Safety Continuum (OpenSSF Memory Safety SIG):
https://memorysafety.openssf.org/memory-safety-continuum/

The Case for Writing Network Drivers in High-Level Programming Languages:
https://arxiv.org/abs/1909.06344

2024 State of Rust Survey:
https://blog.rust-lang.org/2025/02/13/2024-State-Of-Rust-Survey-results/

Most admired and desired programming languages in 2025 StackOverflow Developer Survey:
https://survey.stackoverflow.co/2025/technology/#admired-and-desired