DEFRA
DEFENCE-RELATED RESEARCH ACTION

DEFENSIE
LA DÉFENSE

Defence-related Research Action - DEFRA

ACRONYM: FORCES

Title: FOundations for Reliable, CorrEct, and Secure robotic systems

Duration of the project: 01/12/2024 - 01/03/2029

Key words: Cybersecurity,Memory Safety,Code Transpilation,Defence Robotics,Performance Evaluation, Legacy Code

Budget: 1.789.929 €

of which RHID contribution: 1.663.033 €

PROJECT DESCRIPTION

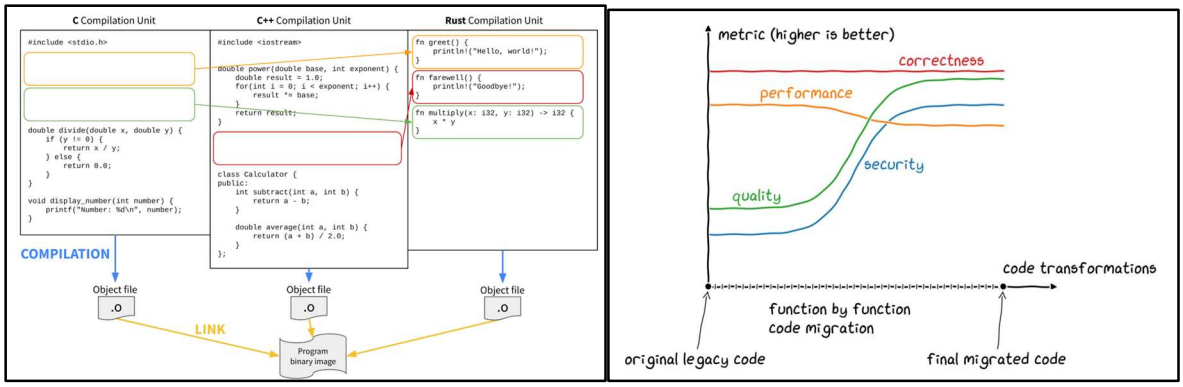
Context

The FORCES project addresses the urgent need for safer and more secure software in cyber-physical defence systems. Legacy programming languages like C and C++ dominate critical defence systems but are inherently prone to memory-related vulnerabilities, posing significant security risks. Modern languages like Rust offer a compelling solution by providing memory safety and concurrency guarantees while maintaining high performance. This transition is crucial for defence robotic systems, which are becoming increasingly important for missions such as reconnaissance and demining. These systems must align with strategic defence plans in cybersecurity to ensure their safety and reliability against evolving cyber threats. By focusing on the migration of legacy systems to Rust, FORCES directly addresses these vulnerabilities and contributes to strengthening the resilience of critical defence infrastructure.

General Objectives

FORCES aims to enhance the security and operational efficiency of defence systems through:

1. The development of **automated, fine-grained code transpilation mechanisms** to facilitate the migration of legacy C/C++ systems to Rust. This method prioritises minimising manual effort and incrementally improving code safety.
2. The creation of a **holistic framework** that includes security analysis, performance benchmarking, and real-world validation, ensuring the migrated systems meet stringent defence requirements.
3. Demonstrating the applicability of these tools and methodologies across diverse defence use cases, ensuring their scalability and adaptability to future needs.



Methodology

The FORCES project is organised into three phases:

1. **Foundational Research (Year 1):** Developing the transpilation methodology and defining metrics for evaluating correctness, security, performance, and maintainability.
2. **Prototype Development (Year 2):** Delivering a TRL 4 prototype of the transpilation tool and validating its functionality on defence-specific scenarios such as robotic systems and Linux kernel drivers.
3. **Refinement and Expansion (Years 3–4):** Refining the tool and evaluation framework to reach TRL 5, applying the methodology to a broader set of use cases, and demonstrating its effectiveness in real-world defence scenarios.

The consortium's structure ensures a cohesive and multidisciplinary approach:

1. **VUB** brings expertise in software languages, code transformation, and performance analysis. It leads the development of the transpilation tool and benchmarks its performance in defence scenarios.
2. **RMA** provides real-world use cases and establishes testbenches for validation. Its focus on military robotics ensures the applicability of the results to realistic operational contexts.
3. **TBE** contributes cybersecurity expertise, defining security metrics and assessing the safety of the transpiled software. It also broadens the project's impact by providing additional defence-related use cases.

Potential Impact on Defence

The project has the potential to transform defence systems by:

1. **Securing Defence Robotics:** Enhancing the safety of robotic systems such as unmanned ground vehicles (UGVs) and robotic arms, reducing risks from memory-related vulnerabilities.
2. **Promoting Innovation:** Equipping defence organisations with advanced tools and methodologies for secure software development.
3. **Cost-Effective Modernisation:** Facilitating the migration of legacy systems to Rust, preserving functionality while addressing vulnerabilities.
4. **Broadening Adoption:** Demonstrating the value of secure programming practices across defence, with potential applications in guidance systems, communication infrastructure, and operational software.

Expected Final Research Results
The project will deliver:

1. **A robust transpilation toolchain** capable of migrating legacy C/C++ code to Rust, ensuring safer and more efficient defence systems.
2. **A validated evaluation framework** with metrics for correctness, security, performance, and maintainability.
3. **Real-world validation** through defence use cases, showcasing improved security and performance of robotic systems.
4. **Knowledge dissemination** via publications in high-impact conferences and journals, as well as workshops and demonstrations for defence stakeholders.
5. **Training and capacity building** for consortium partners, enhancing expertise in cybersecurity and secure coding practices.

Valorisation Perspectives
In the short term, FORCES will deliver tools and methodologies to address immediate cybersecurity challenges in legacy systems. Workshops and publications will raise awareness within the defence community and encourage adoption.

In the medium term, the project's methodologies and tools will find broader applications within defence and beyond, influencing secure programming practices in other critical domains. The toolchain could also serve as the foundation for follow-up projects, scaling the approach to other programming languages and application domains.

By addressing immediate challenges and setting the stage for future advancements, FORCES aligns with strategic defence objectives and demonstrates the value of secure programming for critical infrastructure.

CONTACT INFORMATION

Coordinator
Antonio Paolillo
Vrije Universiteit Brussel (VUB) / Software Languages Lab
antonio.paolillo@vub.be

Partners
Ken Hasselmann
Ecole Royale Militaire - Koninklijke Militaire School (RMA) / RAS-lab
ken.hasselmann@mil.be

Jonathan Pisane
Thales Belgium SA
jonathan.pisane@be.thalesgroup.com

LINK(S)

https://soft.vub.ac.be/forces/
https://mecatron.rma.ac.be/index.php/projects/forces/