Offer #2023-06977

Master internship F/M: Type-based security properties assurance in the Rust-based Redox operating system

**Contract type**: Internship agreement

**Level of qualifications required**: Master's or equivalent

**Fonction**: Internship Research

**About the research centre or Inria department**

The Inria Centre at Rennes University is one of Inria's eight centres and has more than thirty research teams. The Inria Centre is a major and recognized player in the field of digital sciences. It is at the heart of a rich R&D and innovation ecosystem: highly innovative PMEs, large industrial groups, competitiveness clusters, research and higher education players, laboratories of excellence, technological research institute, etc.

**Context**

The internship is proposed in the new cybersecurity research team Sushi focusing on the hardware/software interface. The advisors are operating system and cybersecurity experts from both academia and the french ministry of defense.

https://team.inria.fr/sushi/

Advisors:
- Louis Rilling (DGA, team Sushi) louis.rilling@irisa.fr
- Frédéric Tronel (CentraleSupélec, team Sushi) frederic.tronel@centralesuplec.fr

**Assignment**

**Context:**

Operating systems, especially their kernel, are critical software to build applications aiming at providing security properties. A vulnerability in the kernel opens a door for attackers to bypass the application logic, whatever the correctness of the application itself.

Getting assurance on security properties of operating systems services is known to incur high costs. This traditionally relies on careful design and heavy testing, with costs of up to 400% of a non-secure development effort [1]. While formal verification remains a challenge for this class of software, few projects achieved this goal for an up to 1000% cost [2].

We propose to explore a middle way to get assurance, relying on the compiler to check both of memory safety and higher-level, logical, security properties that should be encoded in the source code using types. It is expected that this reduces the debugging effort and it could make formal verification easier.

**Keywords:** Operating systems; Security; Programming languages; Rust; Redox.

**Main activities**

An example of relevant techniques for operating system development is the typestate design pattern [3]. This technique gives assurance on the implementation of state machines, by assigning a dedicated type to each state of the state machine and encoding state machine transitions as functions converting from a source state to a target state. Invalid transitions are thus made impossible because the converting functions just do not exist. Session types [4] achieve a similar goal for protocols.

Although using types to achieve security has been known for decades, this practice using the programming language of an operating system is hardly explored. Current approaches rely on a separate language and its compiler to write annotations in the original source code and verify that the code satisfies the specified properties [5].

Yet applying typestate analysis on production source code like Linux shows that the security of such development could be improved if types were used to encode logical properties [6]. On the other hand, using the type system of the programming language to ensure functional properties is the topic of ongoing research [7].
The Rust programming language is especially a good candidate as it is strongly- and statically-typed, by design the compiler checks the memory safety of programs, and it is the basis of several open-source operating system projects, including Redox, which explicitly targets security. Moreover, successful Rust-implementations were demonstrated for typestates [8] and session types [9].

One of the mechanisms provided by Redox to achieve security is called schemes. Schemes generalize Unix’ “everything is a file” notion to “everything is a URL”. Using namespace reduction of accessible schemes (the Redox specific setrens system call), open handles can then be used as capabilities [10].

After a study of the state of the art, the suggested work plan is as follows:
- First it should be shown how the typestate design pattern helps getting assurance on isolating clients of a scheme providing access to a cryptographic resource. To this end a representative userspace scheme daemon will be implemented in Rust for Redox.
- Second a comparative study should be done on generically implementing resource read- and write- access restriction in the kernel, using traditional attribute-based checks on one side and type-based segregation of handles on the other side. During the study the two approaches will be implemented and experimentally compared.

References

Skills
Required skills: System programming; curiosity.
Appreciated but not mandated skills: Deep understanding of OSes; Programming in Rust.

Remuneration
Bonus according to current rate

General Information
- Theme/Domain: Security and Confidentiality
Software engineering (BAP E)
- Town/city: Rennes
- Inria Center: Centre Inria de l’Université de Rennes
- Starting date: 2024-02-01
- Duration of contract: 5 months
- Deadline to apply: 2024-01-15

Contacts
- Inria Team: SUSHI (DGD-S)
- Recruiter: Rilling Louis / louis.rilling@irisa.fr
**About Inria**

Inria is the French national research institute dedicated to digital science and technology. It employs 2,600 people. Its 200 agile project teams, generally run jointly with academic partners, include more than 3,500 scientists and engineers working to meet the challenges of digital technology, often at the interface with other disciplines. The Institute also employs numerous talents in over forty different professions. 900 research support staff contribute to the preparation and development of scientific and entrepreneurial projects that have a worldwide impact.

**Warning**: you must enter your e-mail address in order to save your application to Inria. Applications must be submitted online on the Inria website. Processing of applications sent from other channels is not guaranteed.

**Instruction to apply**

**Defence Security**: This position is likely to be situated in a restricted area (ZRR), as defined in Decree No. 2011-1425 relating to the protection of national scientific and technical potential (PPST). Authorisation to enter an area is granted by the director of the unit, following a favourable Ministerial decision, as defined in the decree of 3 July 2012 relating to the PPST. An unfavourable Ministerial decision in respect of a position situated in a ZRR would result in the cancellation of the appointment.

**Recruitment Policy**: As part of its diversity policy, all Inria positions are accessible to people with disabilities.