A propos du centre ou de la direction fonctionnelle

The Inria Rennes - Bretagne Atlantique Centre is one of Inria's eight centres and has more than thirty research teams. The Inria Center 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 SMEs, large industrial groups, competitiveness clusters, research and higher education players, laboratories of excellence, technological research institute, etc.

Contexte et atouts du poste

The candidate will work at Inria/IRISA in the DiverSE team (workplace: University of Rennes 1, Campus de Beaulieu, 35000 Rennes, France).

DiverSE's research is in the area of software engineering, focusing on the management of diversity in the construction of software-intensive systems. The team is actively involved in European, French and industrial projects and is composed of 8 faculty members, 18 PhD students, and 3 engineers.

The candidate can teach but it's not mandatory.

(Cyber)-security is an important topic at Inria/IRISA, with many teams and collaborations (e.g., with DGA). Besides, Rennes is one of the best city to live in France (routinely ranked in the top 3).

Mission confiée

We are looking for highly motivated people in software (in a broad sense) and security. We are studying real-world, popular, open-source, configurable software projects like Linux, Firefox, ffmpeg, VLC, Apache or Hipster – and we hope to have concrete impacts on them!

The candidate should have a strong interest and the skills for performing large-scale experiments with software. In particular:

- advanced software engineering skills, including program analysis, compilation, software automation, software testing. In general we need highly technical software abilities; since software is the main asset we are analyzing;
- experience or interest in software security, for example: fuzzing, code deobfuscation, obfuscation;
- variability and configuration expertise: it is a plus to have knowledge about software configurations, but it's not mandatory.

The candidates will work with two PhD students, at least two professors, and a researcher from DGA.

Principales activités

We expect software systems to deliver the right functionality, as quickly and frugally as possible, in any possible circumstances whatever the hardware, the operating systems, or the input data are.

For fitting such a diversity of settings, most modern systems (operating systems like Linux or Android, Web browsers like Firefox or Chrome, video encoders like ffmpeg, x264 or VLC, mobile and cloud applications, autonomous systems, etc) are subject to variation or come in many variants. Hundreds of configuration options – features, plugins, parameters, hyperparameters – can be combined, each potentially with distinct functionality and effects on time, security, energy consumption, etc.

The ability of varying is therefore a mandatory feature of any (sub-)system. Despite its ubiquitous presence, variability (adaptation at compile-time or run-time) is also an enormous source of complexity with a combinatorial explosion of possible configurations. For example, the Linux kernel exhibits 16000+ options and most options can take "yes", "no", or "module" values. Several works observed that configuration options can be the source of compilation failures, can crash the system, can compute wrong results, or can degrade the performance.

There is another important issue: the surface attack of configurable system is potentially extremely important. Each (combination of) option can be subject to a different attack, for example, an attacker can inspect many configuration-related paths in a program. Overall, we see two fronts:

- configuration can be pointless: users do not need necessarily the ability to configure their systems, especially at run-time. A default configuration can be sufficient for a majority of the use cases while most of the options will simply never been used;
- configuration is a security threat: configuration options cannot only be pointless, they can be unnecessary opportunity and entry point for attackers.

As configuration options can be pointless and represent a threat, our objective is to automatically remove unneeded configuration-related code out of programs. When options are removed, we aim to show that some attacks are no longer possible while the the functionality or usability of the system is not altered. We plan to study real-world systems, attacks, and options to empirically study the effectiveness of our techniques. We also plan to provide some theoretical guarantees about our removal process.

References

- Mathieu Acher, Guillaume Bécan, Benoit Combemale, Benoit Baudry, and Jean-Marc
Compétences
We are looking for highly motivated people in software (in a broad sense) and security. We are studying real-world, popular, open-source, configurable software projects like Linux, Firefox, ffmpeg, VLC, Apache or JHipster – and we hope to have concrete impacts on them!

The candidate should have a strong interest and the skills for performing large-scale experiments with software. In particular:

- advanced software engineering skills, including program analysis, compilation, software automation, software testing. In general we need highly technical software abilities, since software is the main asset we are analyzing;
- experience or interest in software security, for example: fuzzing, code debloating, obfuscation;
- variability and configuration expertise: it is a plus to have knowledge about software configurations, but it’s not mandatory

The candidates will work with two PhD students, at least two professors, and a researcher from DGA.

Avantages
- Subsidized meals
- Partial reimbursement of public transport costs

Rémunération
Post-doc: monthly gross salary amounting to 2653 euros