Formal verification of security protocols: application to contactless systems

Laboratory, institution and university Team EMSEC at IRISA (Rennes).

Salary Funded by the **ERC project POPSTAR**. The salary may vary from 2.350 euros to **2.800 euros** (net per month) depending on the experience of the candidate. The candidate will be an employee of CNRS and enjoy associated benefits (unemployment, retirement and health, etc.). The term of the postdoc is one year, renewal is possible over the whole length of the project (2017-2022).

Contact To apply, send an email to `Stephanie.Delaune@irisa.fr` with your motivations, a brief CV, and possibly some recommandation letters.

Keywords security protocols, formal verification, symbolic model

Context. The shrinking size of microprocessors as well as the ubiquity of wireless communication have led to the proliferation of portable computing devices with novel security requirements. Whereas traditional security protocols achieve their security goals relying solely on cryptographic primitives like encryptions and hash functions, the protocols employed to secure these devices establish and rely in addition on *properties of the physical world*. They may use, as basic building blocks, protocols for ensuring physical proximity, secure localisation, or secure neighbourhood discovery. For instance, physical proximity is an important issue when performing a contactless transaction to ensure that the credit card that will be charged is indeed the one that is close to the reader. Therefore, these protocols of a new kind are now largely deployed to secure many modern applications. There are already 58 millions contactless payment cards in circulation in the UK, and some figures suggest there will be 1 billion contactless payment points on the European continent by 2018. Unfortunately, we often hear about ill-conceived systems, and portable computing devices raise some serious concerns about *privacy*. For instance, you may have heard about security incidents on *contactless payments cards*, and on *keyless systems* that are used nowadays in cars.

Formal verification. One extremely successful approach when designing and analysing security protocols, is the use of formal methods. The purpose of formal verification is to provide rigorous frameworks and techniques to analyse protocols and find their flaws. For example, a flaw has been discovered in the Single-Sign-On protocol used *e.g.* by Google Apps. It has been shown that a malicious application could very easily get access to any other application (*e.g.* Gmail or Google Calendar) of their users [1]. This flaw has been found when analysing the protocol using formal methods, abstracting messages by a term algebra and using the Avantssar validation platform [2]. Another example is a flaw on vote-privacy discovered during the formal and manual analysis of an electronic voting protocol [4]. All these results have been obtained using *formal symbolic models*, where most of the cryptographic details are ignored using abstract structures, and the communication network is assumed to be entirely controlled by an omniscient attacker. The techniques used in symbolic models have become mature and several tools for protocol verification are nowadays available, *e.g.* the Avantssar platform [2], the Tamarin prover [5], and the ProVerif tool [3].

Objectives. The complexity of the verification problem comes from the protocols themselves, as well as the need to clearly state the intended protocol goals and characterise the environment and the attacker capabilities. As experience with traditional protocols has shown, these are highly non-trivial tasks. Many protocols once believed to be secure have been found to be flawed when formally modelled and analysed. In the past three decades, remarkable advances have been made in the automated analysis of standard security protocols, *e.g.* for authentication and key exchange protocols. The time has come to extend these formal models and methods, and to develop new ones, to analyse and secure mobile and wireless communications. This will enable the detection of security flaws, and the development of general design principles to guarantee a high level of security in many applications (*e.g.* payment, identification) that are carried out using contactless devices. Five main tasks (that include several subtasks) have been identified to structure the above goals.

- **Task 1 :** A framework for reasoning about physical properties
- **Task 2 :** Foundations for reasoning about protocols that rely on physical properties
- **Task 3 :** Novel approaches for reasoning about trace equivalence
- **Task 4 :** Automated tool support
- **Task 5 :** Application to real-world contactless systems

We are looking for candidates with good skills in Foundations of Computer Science (logic, automatic deduction, ...) to work on any task of the project. The postdoc will conduct research within the project's goals, and may help in the supervision of student researchers.

Références

[1] A. Armando, R. Carbone, L. Compagna, J. Cuéllar, and M. L. Tobarra. Formal analysis of SAML 2.0 web browser single sign-on : breaking the SAML-based single sign-on for Google apps. In *Proc. 6th ACM Workshop on Formal Methods in Security Engineering (FMSE'08)*, pages 1–10. ACM, 2008.

[2] A. Armando et al. The AVANTSSAR platform for the automated validation of trust and security of service-oriented architectures. In *Proc. 18th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'12)*, volume 7214, pages 267–282. Springer, 2012.

[3] B. Blanchet. An Efficient Cryptographic Protocol Verifier Based on Prolog Rules. In *Proc. 14th Computer Security Foundations Workshop (CSFW'01)*, pages 82–96. IEEE Computer Society Press, 2001.

[4] V. Cortier and B. Smyth. Attacking and fixing Helios : An analysis of ballot secrecy. *Journal of Computer Security*, 21(1) :89–148, 2013.

[5] S. Meier, B. Schmidt, C.J.F. Cremers, and D. Basin. The Tamarin Prover for the Symbolic Analysis of Security Protocols. In *Proc. 25th International Conference on Computer Aided Verification (CAV'13)*, volume 8044 of *Lecture Notes in Computer Science*, pages 696–701. Springer, 2013.