Computer-aided proofs in cryptography: an overview

Gilles Barthe[1], François Dupressoir[1], Benjamin Grégoire[2], Benedikt Schmidt[1], and Pierre-Yves Strub[1]

[1] IMDEA Software Institute, Madrid, Spain
[2] INRIA Sophia-Antipolis Méditerranée, France

The goal of modern cryptography is to design efficient constructions that simultaneously achieve some desired functionality and provable security against resource-bounded adversaries. Over the years, the realm of cryptography has expanded from basic functionalities such as encryption, decryption and key agreement, to elaborate functionalities such as zero-knowledge protocols, secure multiparty computation, and more recently verifiable computation. In many cases, these elaborate functionalities can only be achieved through cryptographic systems, in which several elementary constructions interact. As a consequence of the evolution towards more complex functionalities, cryptographic proofs have become significantly more involved, and more difficult to check. Several cryptographers have therefore advocated the use of tool-supported frameworks for building and verifying proofs; the most vivid recommendation for using computer support is elaborated in a farseeing article [5] in which Shai Halevi describes a potential approach for realizing this vision.

Besides increasing confidence in cryptographic proofs, tool-supported frameworks have the potential to address another prominent difficulty with provable security: because cryptographic proofs are very complex, it is common practice to reason about algorithmic descriptions of the cryptographic constructions, rather than about implementations. As a consequence, several popular implementations of well-known provably secure constructions are vulnerable to attacks! This uncomfortable gap between provable security and cryptographic engineering is the focus of the "real world" security approach that is currently developed, among others, by Paterson and his co-workers [4]. However, we believe that tool support is essential for accomodating the additional complexity introduced by dealing with implementation-level descriptions of cryptographic constructions.

Since 2006, we have been actively working on developing foundations and providing tool support for building and verifying the security of cryptographic constructions. To date, we have constructed several tools, including general frameworks [3, 2] and focused ones that target one specific class of constructions [1]. The talk will provide a brief account of the rationale behind their design and discuss the role of formal proofs in cryptography.

For more information, visit `http://www.easycrypt.info`

References

1. Gilles Barthe, Juan Manuel Crespo, Benjamin Grégoire, César Kunz, Yassine Lakhnech, Benedikt Schmidt, and Santiago Zanella Béguelin. Fully automated analysis of padding-based encryption in the computational model. In *ACM CCS 13: 20th Conference on Computer and Communications Security*, pages 1247–1260. ACM Press, 2013.
2. Gilles Barthe, Benjamin Grégoire, Sylvain Heraud, and Santiago Zanella-Béguelin. Computer-aided security proofs for the working cryptographer. In *Advances in Cryptology – CRYPTO 2011*, volume 6841 of *Lecture Notes in Computer Science*, pages 71–90, Heidelberg, 2011. Springer.
3. Gilles Barthe, Benjamin Grégoire, and Santiago Zanella-Béguelin. Formal certification of code-based cryptographic proofs. In *36th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2009*, pages 90–101, New York, 2009. ACM.
4. Jean Paul Degabriele, Kenneth G. Paterson, and Gaven J. Watson. Provable security in the real world. *IEEE Security & Privacy*, 9(3):33–41, 2011.
5. S. Halevi. A plausible approach to computer-aided cryptographic proofs. Cryptology ePrint Archive, Report 2005/181, 2005.