Automation in computer-aided cryptography: proofs, attacks and designs

Gilles Barthe[1], Benjamin Grégoire[3], César Kunz[1,2], Yassine Lakhnech[4], and Santiago Zanella Béguelin[5]

[1] IMDEA Software Institute
[2] Universidad Politécnica de Madrid
[3] INRIA Sophia Antipolis-Méditerranée
[4] Université de Grenoble, France
[5] Microsoft Research

CertiCrypt [3] and EasyCrypt [2] are machine-checked frameworks for proving the security of cryptographic constructions. Both frameworks adhere to the game-based approach [9, 6, 8] to provable security [7], but revisit its realization from a formal verification pespective. More specifically, CertiCrypt and EasyCrypt use a probabilistic programming language pWHILE for expressing cryptographic constructions, security properties, and computational assumptions, and a probabilistic relational Hoare logic pRHL for justifying reasonings in cryptographic proofs. While both tools coincide in their foundations, they differ in their underlying technologies: CertiCrypt is implemented as a set of libraries in the Coq proof assistant, whereas EasyCrypt uses a verification condition generator for pRHL in combination with off-the-shelf SMT solvers and automated theorem provers. Over the last six years, we have used both tools to verify prominent examples of public-key encryption schemes, modes of operation, signature schemes, hash function designs, zero-knowledge proofs. Recently, we have also used both tools to certify the output of a zero-knowledge compiler [1].

The next challenge is to extend EasyCrypt with automated mechanisms for discovering proofs or attacks. As a first step in this direction, we have developed a front-end that searches for security proofs or attacks for public-key encryption schemes built from one-way trapdoor permutations and random oracles. Given a candidate scheme, the front-end first searches for attacks using a deducibility relation inspired from symbolic cryptography: if an attack is found, it outputs an attacker. If not, the front-end searches for game-based proofs that the scheme is secure: if a proof is found, it outputs a concrete security bound and an EasyCrypt script that can be verified independently. We have evaluated the applicability of the front-end on more than hundred variants of OAEP [5], a widely used padding scheme commonly used for strengthening RSA encryption: pleasingly, it proves most secure variants of OAEP and computes security bounds that match known bounds in many cases. In addition, we have used the front-end in combination with synthesis algorithms to explore the design space of the class of encryption schemes it covers. This has led to the discovery of ZAEP [4], a simplified variant of the OAEP padding scheme that can be used to strengthen RSA encryption with exponents 2 and 3.

More information about the project can be found at:

http://easycrypt.gforge.inria.fr

References

1. José Bacelar Almeida, Manuel Barbosa, Endre Bangerter, Gilles Barthe, Stephan Krenn, and Santiago Zanella Béguelin. Full proof cryptography: Verifiable compilation of efficient zero-knowledge protocols. In *19th ACM Conference on Computer and Communications Security, CCS 2012*. ACM, 2012.
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. Gilles Barthe, David Pointcheval, and Santiago Zanella Béguelin. Verified security of redundancy-free encryption from Rabin and RSA. In *19th ACM Conference on Computer and Communications Security, CCS 2012*. ACM, 2012. To appear.
5. Mihir Bellare and Phillip Rogaway. Optimal asymmetric encryption. In *Advances in Cryptology – EUROCRYPT 1994*, volume 950 of *Lecture Notes in Computer Science*, pages 92–111, Heidelberg, 1994. Springer.
6. Mihir Bellare and Phillip Rogaway. The security of triple encryption and a framework for code-based game-playing proofs. In *Advances in Cryptology – EUROCRYPT 2006*, volume 4004 of *Lecture Notes in Computer Science*, pages 409–426, Heidelberg, 2006. Springer.
7. Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28(2):270–299, 1984.
8. S. Halevi. A plausible approach to computer-aided cryptographic proofs. Cryptology ePrint Archive, Report 2005/181, 2005.
9. Victor Shoup. Sequences of games: a tool for taming complexity in security proofs. Cryptology ePrint Archive, Report 2004/332, 2004.