Formal Models and Techniques for Analyzing Security Protocols: A Tutorial

Véronique Cortier
LORIA, CNRS
cortier@loria.fr

Steve Kremer
INRIA Nancy
steve.kremer@inria.fr

the essence of knowledge
Boston — Delft

Foundations and Trends® in Programming Languages

Published, sold and distributed by:
now Publishers Inc.
PO Box 1024
Hanover, MA 02339
United States
Tel. +1-781-985-4510
www.nowpublishers.com
sales@nowpublishers.com

Outside North America:
now Publishers Inc.
PO Box 179
2600 AD Delft
The Netherlands
Tel. +31-6-51115274

The preferred citation for this publication is

V. Cortier and S. Kremer. *Formal Models and Techniques for Analyzing Security Protocols: A Tutorial*. Foundations and Trends® in Programming Languages, vol. 1, no. 3, pp. 151–267, 2014.

This Foundations and Trends® issue was typeset in LaTeX using a class file designed by Neal Parikh. Printed on acid-free paper.

ISBN: 978-1-60198-903-1

Foundations and Trends® in Programming Languages
Volume 1, Issue 3, 2014

Editorial Scope

Topics

Foundations and Trends® in Programming Languages publishes survey and tutorial articles in the following topics:

- Abstract interpretation
- Compilation and interpretation techniques
- Domain specific languages
- Formal semantics, including lambda calculi, process calculi, and process algebra
- Language paradigms
- Mechanical proof checking
- Memory management
- Partial evaluation
- Program logic
- Programming language implementation
- Programming language security
- Programming languages for concurrency
- Programming languages for parallelism
- Program synthesis
- Program transformations and optimizations
- Program verification
- Runtime techniques for programming languages
- Software model checking
- Static and dynamic program analysis
- Type theory and type systems

Information for Librarians

Foundations and Trends® in Programming Languages, 2014, Volume 1, 4 issues. ISSN paper version 2325-1107. ISSN online version 2325-1131. Also available as a combined paper and online subscription.

Foundations and Trends® in Programming Languages
Vol. 1, No. 3 (2014) 151–267

DOI: 10.1561/2500000001

Formal Models and Techniques for Analyzing Security Protocols: A Tutorial

Véronique Cortier
LORIA, CNRS
cortier@loria.fr

Steve Kremer
INRIA Nancy
steve.kremer@inria.fr

Contents

Abstract

Security protocols are distributed programs that aim at securing communications by the means of cryptography. They are for instance used to secure electronic payments, home banking and more recently electronic elections. Given the financial and societal impact in case of failure, and the long history of design flaws in such protocols, formal verification is a necessity. A major difference from other safety critical systems is that the properties of security protocols must hold in the presence of an arbitrary adversary. The aim of this paper is to provide a tutorial to some modern approaches for formally modeling protocols, their goals and automatically verifying them.

V. Cortier and S. Kremer. *Formal Models and Techniques for Analyzing Security Protocols: A Tutorial*. Foundations and Trends® in Programming Languages, vol. 1, no. 3, pp. 151–267, 2014.
DOI: 10.1561/2500000001.

1

Introduction

Security protocols are used to protect electronic transactions. The probably most used security protocol is the SSL/TLS protocol which underlies the https protocol in web browsers. It may be used for electronic commerce, or simply to encrypt web search queries on their way between the host and the search engine. There are of course many other protocols in use, e.g. to authenticate to providers on mobile phones or withdraw cash on an ATM. Moreover, the digitalization of our modern society requires the use of security protocols in an increasing number of contexts, such as electronic passports that may include RFID chips, electronic elections that allow for Internet voting, etc.

We may think of security protocols as distributed programs that make use of cryptography, e.g. encryption, to achieve a security property, such as confidentiality of some data, e.g. your credit card number. Given the difficulty of designing correct distributed systems in general, it is not surprising that many flaws were discovered in security protocols, even without breaking the underlying cryptography. During the last 30 years many research efforts were spent on designing techniques and tools to analyze security protocols. One may trace this line of work back to the seminal work of Dolev and Yao [1981] who pioneered the ideas of an attacker who completely controls the communication network, has an unbounded computational power, but manipulates protocol messages according to some predefined rules, idealizing the protections offered by cryptography. These techniques not only allowed to better understand the principles underlying secure protocol design, but also resulted in mature tools, for automated protocol analysis, and the discovery of many attacks. For example, while designing a formal model of Google's Single Sign-On protocol, that allows a user to identify himself only once and then access various applications (such as Gmail or Google calendar), Armando et al. [2008] discovered that a dishonest service provider could impersonate any of its users at another service provider. This flaw has been corrected since. Basin et al. [2012] have identified flaws and proposed fixes for the ISO/IEC 9798 standard for entity authentication, using automated protocol verification tools. The standard has been revised to include their proposed amendments. Bortolozzo et al. [2010] designed a dedicated analysis tool for hardware security tokens that implement the PKCS#11 standard. The tool automatically reverse-engineers the tokens to extract its configuration, builds an abstract model to be analyzed and verifies the attack on the token if an attack is found. They were able to find unknown attacks on more than 10 commercial tokens.

This paper proposes a tutorial, presenting modern techniques to model and automatically analyze security protocols. Given the large body of work in this area we do not aim to be exhaustive and only present some selected methods and results. We expect that this tutorial could serve as a basis for a master, or graduate course, or allow researchers from different areas to get an overview of the kinds of techniques that are used. The outline of the tutorial is as follows.

- We first present an informal description of our running example, the Needham Schroeder public key protocol that we used for illustration purposes in the remainder of the paper.

- Then, we explain how protocol messages can be modeled as first order terms, and how adversary capabilities can be modeled by an inference system. We also provide a decision algorithm for deduction, i.e. the adversary's capability to construct new messages.

- Next, we introduce a more general model, based on equational theories. We revisit deduction and define a notion of message indistinguishability, called static equivalence. We again provide a decision procedure for static equivalence for a simple equational theory representing symmetric encryption.

- We continue by introducing a process calculus, the applied pi calculus, which we use to model protocols. One of the main differences with the original pi calculus is that the calculus allows communication of messages represented by terms, rather than only names. We illustrate how protocols can be conveniently modeled in this formalism.

- Next we discuss how we can express security properties of protocols modelled in the applied pi calculus. We cover different flavors of confidentiality, authentication, but also anonymity properties, expressed as behavioral equivalences of processes.

- We go on discussing automated verification. We first consider the case when protocol participants only execute a bounded number of sessions. We present a decision procedure based on constraint solving which allows to decide secrecy in this setting.

- Finally, we show that the general case, where the number of sessions is unbounded, is undecidable. We show that nevertheless it is possible to design tools that are able to analyze protocols. This comes at the cost that termination is not guaranteed. In particular, we present an approach based on a representation of the protocol and the adversary as Horn clauses and describe a resolution based procedure implemented in the ProVerif tool.

- We conclude the tutorial by briefly discussing some other approaches for automated verification and other directions in this research area.

References

M. Abadi. Secrecy by typing in security protocols. In *Proc. 3rd International Symposium on Theoretical Aspects of Computer Software (TACS'97)*, volume 1281 of *Lecture Notes in Computer Science*, pages 611–638. Springer, 1997.

M. Abadi and V. Cortier. Deciding knowledge in security protocols under equational theories. In *Proc. 31st International Colloquium on Automata, Languages, and Programming (ICALP'04)*, volume 3142 of *Lecture Notes in Computer Science*, pages 46–58. Springer, 2004.

M. Abadi and V. Cortier. Deciding knowledge in security protocols under equational theories. *Theoretical Computer Science*, 387(1-2):2–32, 2006.

M. Abadi and C. Fournet. Mobile values, new names, and secure communication. In *Proc. 28th ACM Symposium on Principles of Programming Languages (POPL'01)*, pages 104–115. ACM, 2001.

M. Abadi and C. Fournet. Private authentication. *Theoretical Computer Science*, 322(3):427–476, 2004.

M. Abadi and A. D. Gordon. A calculus for cryptographic protocols: The spi calculus. *Information and Computation*, 148(1):1–70, 1999.

M. Abadi and P. Rogaway. Reconciling two views of cryptography (the computational soundness of formal encryption). *Journal of Cryptology*, 15(2): 103–127, 2002.

R. Amadio and W. Charatonik. On name generation and set-based analysis in the Dolev-Yao model. In *Proc. 13th International Conference on Concurrency Theory (CONCUR'02)*, Lecture Notes in Computer Science, pages 499–514. Springer, 2002.

R. Amadio and D. Lugiez. On the reachability problem in cryptographic protocols. In *Proc. 12th International Conference on Concurrency Theory (CONCUR'00)*, volume 1877 of *Lecture Notes in Computer Science*, pages 380–394, 2000.

R. Amadio, D. Lugiez, and V. Vanackère. On the symbolic reduction of processes with cryptographic functions. *Theoretical Computer Science*, 290 (1):695–740, 2002.

S. Anantharaman, P. Narendran, and M. Rusinowitch. Intruders with caps. In *Proc. 18th International Conference on Term Rewriting and Applications (RTA'07)*, volume 4533 of *Lecture Notes in Computer Science*, pages 20–35. Springer, 2007.

M. Arapinis, T. Chothia, E. Ritter, and M. D. Ryan. Analysing unlinkability and anonymity using the applied pi calculus. In *Proc. 23rd Computer Security Foundations Symposium (CSF'10)*, pages 107–121. IEEE Comp. Soc. Press, 2010.

M. Arapinis, E. Ritter, and M. D. Ryan. Statverif: Verification of stateful processes. In *Proc. 24th IEEE Computer Security Foundations Symposium (CSF'11)*, pages 33–47. IEEE Comp. Soc. Press, 2011.

A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuellar, P. Hankes Drielsma, P.-C. Héam, O. Kouchnarenko, J. Mantovani, S. Mödersheim, D. von Oheimb, M. Rusinowitch, J. Santiago, M. Turuani, L. Viganò, and L. Vigneron. The AVISPA Tool for the automated validation of internet security protocols and applications. In K. Etessami and S. Rajamani, editors, *17th International Conference on Computer Aided Verification, CAV'2005*, volume 3576 of *Lecture Notes in Computer Science*, pages 281–285, Edinburgh, Scotland, 2005. Springer.

A. Armando, R. Carbone, L. Compagna, J. Cuellar, and L. Tobarra Abad. 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 2008)*, pages 1–10, 2008.

F. Baader and T. Nipkow. *Term rewriting and all that.* Cambridge University Press, 1998. ISBN 978-0-521-45520-6.

F. Baader and W. Snyder. Unification theory. In J. A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, pages 445–532. Elsevier and MIT Press, 2001. ISBN 0-444-50813-9, 0-262-18223-8.

M. Backes, B. Pfitzmann, and M. Waidner. The reactive simulatability (RSIM) framework for asynchronous systems. *Information and Computation*, 205(12):1685–1720, 2007.

G. Barthe, B. Grégoire, S. Heraud, and S. 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. Springer, 2011.

D. Basin, C. Cremers, and S. Meier. Provably repairing the ISO/IEC 9798 standard for entity authentication. In *Proc. 1st Conference on Principles of Security and Trust (POST'12)*, volume 7215 of *Lecture Notes in Computer Science*, pages 129–148. Springer, 2012.

M. Baudet. Deciding security of protocols against off-line guessing attacks. In *Proc. 12th ACM Conference on Computer and Communications Security (CCS'05)*, pages 16–25. ACM, November 2005.

M. Baudet, V. Cortier, and S. Delaune. YAPA: A generic tool for computing intruder knowledge. *ACM Transactions on Computational Logic*, 14, 2013.

J. Bengtson, K. Bhargavan, C. Fournet, A. D. Gordon, and S. Maffeis. Refinement types for secure implementations. *ACM Trans. Program. Lang. Syst.*, 33(2):8:1–8:45, 2011.

M. Berrima, N. Ben Rajeb, and V. Cortier. Deciding knowledge in security protocols under some e-voting theories. *Theoretical Informatics and Applications (RAIRO-ITA)*, 45:269–299, 2011.

B. Blanchet. An efficient cryptographic protocol verifier based on prolog rules. In *Proc. of the 14th Computer Security Foundations Workshop (CSFW'01)*. IEEE Comp. Soc. Press, 2001.

B. Blanchet. Automatic Proof of Strong Secrecy for Security Protocols. In *Proc. Symposium on Security and Privacy (SP'04)*, pages 86–100. IEEE Comp. Soc. Press, 2004.

B. Blanchet. A computationally sound mechanized prover for security protocols. In *Proc. IEEE Symposium on Security and Privacy (SP'06)*, pages 140–154. IEEE Comp. Soc. Press, 2006.

B. Blanchet. Automatic verification of correspondences for security protocols. *Journal of Computer Security*, 17(4):363–434, 2009.

B. Blanchet. *Formal Models and Techniques for Analyzing Security Protocols*, chapter Using Horn Clauses for Analyzing Security Protocols. Volume 5 of Cortier and Kremer [2011], 2011.

B. Blanchet and M. Paiola. Automatic verification of protocols with lists of unbounded length. In *Proc. 17th ACM Conference on Computer and Communications Security (CCS'10)*, pages 573–584. ACM, 2010.

B. Blanchet, M. Abadi, and C. Fournet. Automated Verification of Selected Equivalences for Security Protocols. In *Proc. 20th Symposium on Logic in Computer Science (LICS'05)*, pages 331–340. IEEE Comp. Soc. Press, 2005.

B. Blanchet, B. Smyth, and V. Cheval. *ProVerif 1.88: Automatic Cryptographic Protocol Verifier, User Manual and Tutorial*, 2013.

F. Böhl and D. Unruh. Symbolic universal composability. In *Proc. 26rd Computer Security Foundations Symposium (CSF'13)*, pages 257–271, 2013.

M. Bond and R. Anderson. API level attacks on embedded systems. *IEEE Computer Magazine*, pages 67–75, October 2001.

M. Bortolozzo, M. Centenaro, R. Focardi, and G. Steel. Attacking and fixing PKCS#11 security tokens. In *Proc. 17th ACM Conference on Computer and Communications Security (CCS'10)*, pages 260–269. ACM, 2010.

M. Brusò, K. Chatzikokolakis, and J. den Hartog. Formal verification of privacy for rfid systems. In *Proc. 23rd Computer Security Foundations Symposium (CSF'10)*, pages 75–88. IEEE Comp. Soc. Press, 2010.

S. Bursuc, H. Comon-Lundh, and S. Delaune. Associative-commutative deducibility constraints. In *Proc. 24th Annual Symposium on Theoretical Aspects of Computer Science (STACS'07)*, volume 4393 of *Lecture Notes in Computer Science*, pages 634–645. Springer, 2007.

D. Cadé and B. Blanchet. Proved generation of implementations from computationally-secure protocol specifications. In *Proc. 2nd Conference on Principles of Security and Trust (POST'13)*, volume 7796 of *Lecture Notes in Computer Science*, pages 63–82. Springer, 2013.

R. Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *Proc. 42nd IEEE Symp. on Foundations of Computer Science (FOCS'01)*, pages 136–145. IEEE Comp. Soc. Press, 2001.

R. Chadha, Ş. Ciobâcă, and S. Kremer. Automated verification of equivalence properties of cryptographic protocols. In *Programming Languages and Systems —Proc. 21th European Symposium on Programming (ESOP'12)*, volume 7211 of *Lecture Notes in Computer Science*, pages 108–127. Springer, 2012.

S. Chaki and A. Datta. Aspier: An automated framework for verifying security protocol implementations. In *Proc. 22nd Computer Security Foundations Symposium (CSF'09)*, pages 172–185. IEEE Comp. Soc. Press, 2009.

V. Cheval. *Automatic verification of cryptographic protocols: privacy-type properties.* Thèse de doctorat, Laboratoire Spécification et Vérification, ENS Cachan, France, December 2012.

V. Cheval. Apte: an algorithm for proving trace equivalence. In *Proc. 20th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'14)*, volume 8413 of *Lecture Notes in Computer Science*, pages 587–592. Springer, 2014.

V. Cheval, H. Comon-Lundh, and S. Delaune. Trace equivalence decision: Negative tests and non-determinism. In *Proc. 18th ACM Conference on Computer and Communications Security (CCS'11)*, pages 321–330. ACM, 2011.

V. Cheval, V. Cortier, and A. Plet. Lengths may break privacy – or how to check for equivalences with length. In *Proc. 25th International Conference on Computer Aided Verification (CAV'13)*, volume 8043 of *Lecture Notes in Computer Science*, pages 708–723. Springer, 2013.

Y. Chevalier and M. Rusinowitch. Compiling and securing cryptographic protocols. *Inf. Process. Lett.*, 110(3):116–122, 2010.

T. Chothia and V. Smirnov. A traceability attack against e-passports. In *Proc. 14th International Conference on Financial Cryptography and Data Security (FC'10)*, volume 6052 of *Lecture Notes in Computer Science*, pages 20–34. Springer, 2010.

N. Chridi, M. Turuani, and M. Rusinowitch. Decidable analysis for a class of cryptographic group protocols with unbounded lists. In *Proc. 22nd Computer Security Foundations Symposium (CSF'09)*, pages 277–289. IEEE Comp. Soc. Press, 2009.

Ş. Ciobâcă, S. Delaune, and S. Kremer. Computing knowledge in security protocols under convergent equational theories. *Journal of Automated Reasoning*, 48(2):219–262, 2012.

H. Comon-Lundh and V. Shmatikov. Intruder deductions, constraint solving and insecurity decision in presence of Exclusive Or. In *Proc. 18th Annual IEEE Symposium on Logic in Computer Science (LICS '03)*, pages 271–280, Los Alamitos, CA, 2003. IEEE Computer Society.

H. Comon-Lundh, V. Cortier, and E. Zălinescu. Deciding security properties for cryptographic protocols. Application to key cycles. *ACM Transactions on Computational Logic*, 11(2), 2010.

R. Corin and S. Etalle. An improved constraint-based system for the verification of security protocols. In *Proc. 9th International Static Analysis Symposium (SAS'02)*, volume 2477 of *Lecture Notes in Computer Science*, pages 326–341. Springer, 2003.

R. Corin, J. Doumen, and S. Etalle. Analysing password protocol security against off-line dictionary attacks. *ENTCS*, 121:47–63, 2005.

R. Corin, S. Etalle, and A. Saptawijaya. A logic for constraint-based security protocol analysis. In *Proc. IEEE Symposium on Security and Privacy (SP'06)*, pages 155–168. IEEE Comp. Soc. Press, 2006.

V. Cortier and S. Delaune. Decidability and combination results for two notions of knowledge in security protocols. *Journal of Automated Reasoning*, 48, 2012.

V. Cortier and S. Kremer, editors. *Formal Models and Techniques for Analyzing Security Protocols*, volume 5 of *Cryptology and Information Security Series*. IOS Press, 2011.

V. Cortier, S. Kremer, and B. Warinschi. A survey of symbolic methods in computational analysis of cryptographic systems. *Journal of Automated Reasoning*, 46(3-4):225–259, 2010.

C. Cremers. The Scyther Tool: Verification, falsification, and analysis of security protocols. In *Proc. 20th International Conference on Computer Aided Verification (CAV'08)*, volume 5123 of *Lecture Notes in Computer Science*, pages 414–418. Springer, 2008.

M. D. Davis and E. J. Weyuker. *Computability, complexity and languages*, chapter 7, pages 128–132. Computer Science and Applied Mathematics. Academic Press, 1983.

H. de Nivelle. *Ordering Refinements of Resolution*. PhD thesis, Technische Universiteit Delft, 1995.

S. Delaune and F. Jacquemard. A decision procedure for the verification of security protocols with explicit destructors. In *Proc. 11th ACM Conference on Computer and Communications Security (CCS'04)*, pages 278–287. ACM, 2004.

S. Delaune and F. Jacquemard. Decision procedures for the security of protocols with probabilistic encryption against offline dictionary attacks. *Journal of Automated Reasoning*, 36(1-2):85–124, 2006.

S. Delaune, S. Kremer, and O. Pereira. Simulation based security in the applied pi calculus. In *Proc. 29th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'09)*, volume 4 of *Leibniz International Proceedings in Informatics*, pages 169–180. Leibniz-Zentrum für Informatik, 2009a.

S. Delaune, S. Kremer, and M. D. Ryan. Verifying privacy-type properties of electronic voting protocols. *Journal of Computer Security*, 17(4):435–487, 2009b.

S. Delaune, S. Kremer, and M. D. Ryan. Symbolic bisimulation for the applied pi calculus. *Journal of Computer Security*, 18(2):317–377, 2010a.

S. Delaune, S. Kremer, and G. Steel. Formal analysis of PKCS#11 and proprietary extensions. *Journal of Computer Security*, 18(6):1211–1245, 2010b.

S. Delaune, S. Kremer, M. D. Ryan, and G. Steel. Formal analysis of protocols based on TPM state registers. In *Proc. 24th IEEE Computer Security Foundations Symposium (CSF'11)*, pages 66–82. IEEE Press, 2011.

S. Delaune, S. Kremer, and D. Pasaila. Security protocols, constraint systems, and group theories. In *Proc. 6th International Joint Conference on Automated Reasoning (IJCAR'12)*, volume 7364 of *Lecture Notes in Artificial Intelligence*, pages 164–178. Springer, 2012.

W. Diffie and M. Helman. New directions in cryptography. *IEEE Transactions on Information Society*, 22(6):644–654, 1976.

D. Dolev and A.C. Yao. On the security of public key protocols. In *Proc. 22nd Symposium on Foundations of Computer Science*, pages 350–357. IEEE Comp. Soc. Press, 1981.

H. Dong, N.and Jonker and J. Pang. Analysis of a receipt-free auction protocol in the applied pi calculus. In *Proc. International Workshop on Formal Aspects in Security and Trust (FAST'10)*, volume 6561 of *Lecture Notes in Computer Science*, pages 223–238. Springer, 2011.

J. Dreier, P. Lafourcade, and Y. Lakhnech. Formal verification of e-auction protocols. In *Proc. 2nd Conferences on Principles of Security and Trust (POST'13)*, volume 7796 of *Lecture Notes in Computer Science*, pages 247–266. Springer, 2013.

F. Dupressoir, A. D. Gordon, J. Jürjens, and D. A. Naumann. Guiding a general-purpose c verifier to prove cryptographic protocols. In *Proc. 24th Computer Security Foundations Symposium (CSF'11)*, pages 3–17. IEEE Comp. Soc. Press, 2011.

N. Durgin, P. Lincoln, J. Mitchell, and A. Scedrov. Undecidability of bounded security protocols. In *Proc. Workshop on Formal Methods and Security Protocols*, 1999.

S. Escobar, C. Meadows, and J. Meseguer. Maude-npa: Cryptographic protocol analysis modulo equational properties. In *Foundations of Security Analysis and Design V*, volume 5705 of *Lecture Notes in Computer Science*, pages 1–50. Springer, 2009.

R. Focardi and M. Maffei. *Formal Models and Techniques for Analyzing Security Protocols*, chapter Types for Security Protocols. Volume 5 of Cortier and Kremer [2011], 2011.

R. Focardi and F. Martinelli. A uniform approach for the definition of security properties. In *Proc. World Congress on Formal Methods (FM'99)*, Lecture Notes in Computer Science, pages 794–813. Springer, 1999.

A. Fujioka, T. Okamoto, and K. Ohta. A practical secret voting scheme for large scale elections. In *Advances in Cryptology - AUSCRYPT'92*, volume 718 of *Lecture Notes in Computer Science*, pages 244–251. Springer-Verlag, 1992.

Th. Genet and F. Klay. Rewriting for cryptographic protocol verification. In *Proc. 17th International Conference on Automated Deduction (CADE'00)*, volume 1831 of *Lecture Notes in Computer Science*, pages 271–290. Springer, 2000.

S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28, 1984.

D. Gollmann. What do we mean by entity authentication? In *Proc. Symposium on Security and Privacy (SP'96)*, pages 46–54. IEEE Comp. Soc. Press, 1996.

J. Goubault-Larrecq. A method for automatic cryptographic protocol verification (extended abstract). In *Proc. Workshops of the 15th International Parallel and Distributed Processing Symposium*, volume 1800 of *Lecture Notes in Computer Science*, pages 977–984. Springer, 2000.

J. Goubault-Larrecq and F. Parrennes. Cryptographic protocol analysis on real C code. In *Proc. 6th International Conference on Verification, Model Checking and Abstract Interpretation (VMCAI'05)*, volume 3385 of *Lecture Notes in Computer Science*, pages 363–379. Springer, 2005.

ISO/IEC-9798-1. *Information technology - Security techniques - Entity authentication mechanisms; Part 1: General model. ISO/IEC 9798-1.* International Organization for Standardization, second edition edition, sep 1991.

F. Jacquemard, M.Rusinowitch, and L. Vigneron. Compiling and verifying security protocols. In *Proc. 7th International Conference on Logic for Programming and Automated Reasoning (LPAR'00)*, volume 1955 of *Lecture Notes in Computer Science*, pages 131–160. Springer, 2000.

S. Kremer and R. Künnemann. Automated analysis of security protocols with global state. In *Proceedings of the 35th IEEE Symposium on Security and Privacy (SP'14)*, pages 163–178. IEEE Comp. Soc. Press, 2014.

S. Kremer and L. Mazaré. Adaptive soundness of static equivalence. In *Proc. 12th European Symposium on Research in Computer Security (ESORICS'07)*, volume 4734 of *Lecture Notes in Computer Science*, pages 610–625. Springer, 2007.

S. Kremer and M. D. Ryan. Analysis of an electronic voting protocol in the applied pi-calculus. In *Programming Languages and Systems — Proc. 14th European Symposium on Programming (ESOP'05)*, volume 3444 of *Lecture Notes in Computer Science*, pages 186–200. Springer, 2005.

S. Kremer, A. Mercier, and R. Treinen. Reducing equational theories for the decision of static equivalence. *Journal of Automated Reasoning*, 48(2): 197–217, 2012.

R. Küsters, T. Truderung, and J. Graf. A framework for the cryptographic verification of java-like programs. In *Proc. 25th Computer Security Foundations Symposium (CSF'12)*, pages 198–212. IEEE Comp. Soc. Press, 2012.

D. Longley and S. Rigby. An automatic search for security flaws in key management schemes. *Computers and Security*, 11(1):75–89, March 1992.

G. Lowe. Breaking and fixing the Needham-Schroeder public-key protocol using FDR. In *Proc. 2nd International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'96)*, volume 1055 of *Lecture Notes in Computer Science*, pages 147–166. Springer-Verlag, 1996.

G. Lowe. A hierarchy of authentication specifications. In *Proc. 10th Computer Security Foundations Workshop (CSFW'97)*, pages 31–44. IEEE Comp. Soc. Press, 1997a.

G. Lowe. Casper: a compiler for the analysis of security protocols. In *Proc. 10th Computer Security Foundations Workshop (CSFW'97)*, pages 18–30. IEEE Comp. Soc. Press, 1997b.

G. Lowe. Analysing protocols subject to guessing attacks. *Journal of Computer Security*, 12(1):83–98, 2004.

C. Meadows. The NRL protocol analyzer: An overview. *Journal of Logic Programming*, 26(2):113–131, 1996.

J. Millen. A necessarily parallel attack. In *FMSP '99*, 1999.

J. Millen and G. Denker. Capsl and mucapsl. *J. Telecommunications and Information Technology*, 4:16–27, 2002.

J. Millen and V. Shmatikov. Constraint solving for bounded-process cryptographic protocol analysis. In *Proc. 8th ACM Conference on Computer and Communications Security (CCS'01)*, 2001.

J. C. Mitchell, M. Mitchell, and U. Stern. Automated analysis of cryptographic protocols using Murφ. In *Proc. IEEE Symposium on Security and Privacy (SP'97)*, pages 141–153, 1997.

S. Mödersheim. Abstraction by set-membership: verifying security protocols and web services with databases. In *Proc. 17th ACM Conference on Computer and Communications Security (CCS'10)*, pages 351–360. ACM, 2010.

D. Monniaux. Abstracting cryptographic protocols with tree automata. *Sci. Comput. Program.*, 47(2-3):177–202, 2003.

R. M. Needham and M. D. Schroeder. Using encryption for authentication in large networks of computers. *Communications of the ACM*, 21(12):993–999, 1978.

L. C. Paulson. Mechanized proofs for a recursive authentication protocol. In *Proc. 10th Computer Security Foundations Workshop (CSFW'97)*, pages 84–95. IEEE Comp. Soc. Press, 1997.

L. C. Paulson. The inductive approach to verifying cryptographic protocols. *Journal of Computer Security*, 6(1/2):85–128, 1998.

Alfredo Pironti and Riccardo Sisto. Provably correct Java implementations of Spi Calculus security protocols specifications. *Computers & Security*, 29: 302–314, 2010.

M. Rusinowitch and M. Turuani. Protocol insecurity with finite number of sessions is NP-complete. In *Proc. 14th Computer Security Foundations Workshop (CSFW'01)*, pages 174–190. IEEE Comp. Soc. Press, 2001.

M. Rusinowitch and M. Turuani. Protocol Insecurity with Finite Number of Sessions and Composed Keys is NP-complete. *Theoretical Computer Science*, 299:451–475, 2003.

P.Y.A Ryan, S.A. Schneider, M. Goldsmith, G. Lowe, and A.W. Roscoe. *Modelling and Analysis of Security Protocols*. Addison Wesley, 2000.

B. Schmidt, S. Meier, C. Cremers, and D. Basin. Automated analysis of Diffie-Hellman protocols and advanced security properties. In *Proc. 25th IEEE Computer Security Foundations Symposium (CSF'12)*, pages 78–94. IEEE Comp. Soc. Press, 2012.

B. Schmidt, S. Meier, C. 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.

S. Schneider. Verifying authentication protocols with CSP. In *Proc. 10th Computer Security Foundations Workshop (CSFW'97)*. IEEE Comp. Soc. Press, 1997.

D. Song. Athena, an automatic checker for security protocol analysis. In *Proc. 12th IEEE Computer Security Foundations Workshop (CSFW'99)*. IEEE Comp. Soc. Press, 1999.

J. Thayer, J. Herzog, and J. Guttman. Strand spaces: proving security protocols correct. *IEEE Journal of Computer Security*, 7:191–230, 1999.

A. Tiu and J. Dawson. Automating open bisimulation checking for the spi-calculus. In *Proc. 23rd Computer Security Foundations Symposium (CSF'10)*, pages 307–321. IEEE Comp. Soc. Press, 2010.

T. Truderung. Selecting theories and recursive protocols. In *Proc. 16th International Conference on Concurrency Theory (Concur'05)*, volume 3653 of *Lecture Notes in Computer Science*, pages 217–232. Springer, 2005.

Ch. Weidenbach. Towards an automatic analysis of security protocols in first-order logic. In *Proc. 16th International Conference on Automated Deduction (CADE'99)*, volume 1632 of *Lecture Notes in Computer Science*, pages 314–328. Springer, 1999.

T.Y.C. Woo and S.S. Lam. Authentication for distributed systems. In *Proc. Symposium on Security and Privacy (SP'92)*, pages 178–194. IEEE Comp. Soc. Press, 1992.