Security-Oriented Formal Techniques

Marcantoni F. , Paoloni F. and Polzonetti A.
School of Science and Technology – University of Camerino ITALY

Abstract - *Please consider these Instructions as guidelines for preparation of Final Camera-ready Papers. The Camera-Ready Papers would be acceptable as long as it is formatted reasonably close to the format being suggested here. Note that these instructions are reasonably comparable to the standard IEEE typesetting format. Type the abstract (100 words minimum and 150 words maximum) using Italic font with point size 10. The abstract is an essential part of the paper. Use short, direct, and complete sentences. It should be brief and as concise as possible*.

Keywords: Security, Formal Methods

1 Introduction

Security of software systems is a critical issue in a world where Information Technology is becoming more and more pervasive. The number of services for everyday life that are provided via electronic networks is rapidly increasing, as witnessed by the longer and longer list of words with the prefix "e", such as e-banking, e-commerce, e-government, where the "e" substantiates their electronic nature. These kinds of services usually require the exchange of sensible data and the sharing of computational resources, thus needing strong security requirements because of the relevance of the exchanged information and the very distributed and untrusted environment, the Internet, in which they operate. It is important, for example, to ensure the authenticity and the secrecy of the exchanged messages, to establish the identity of the involved entities, and to have guarantees that the different system components correctly interact, without violating the required global properties.

Unfortunately, many authoritative security-related organizations as, e.g., the CERT at Carnegie Mellon University, report a growing number of computer system vulnerabilities which are often the result of exploits against defects in the design or code of software. The approach most commonly employed to address such defects is to attempt to a posteriori "repair the flaw" by making it more difficult for those defects to be exploited. This solution, however, does not certainly get to the root cause of the problem and threat. A complementary approach is, instead, to model and verify security requirements from the very first specification of software systems, so to reduce as much as possible the presence of vulnerabilities on the final product. The use of formal techniques can thus play an important role to reveal possible security flaws from the very first phases of software development, to understand in depth the causes, and to remove them before it is too late and it becomes necessary to invent, if possible, some retroactive remedy. The interest in formal methods for security is confirmed by a very active international community, and by the increasing number of new international workshops and conferences on the topic.

The aim of this project is to put together a consortium of 3 Universities which are already active in the fields of formal methods for security and of software and protocol verification, and which will focus their effort on common research targets. We intend to broadly work on many different aspects of security, mainly focusing on "language-based" techniques, which have the advantage of verifying security of programs directly on their formal specification, without the need of analysing their execution. We believe that this approach is particularly appealing both because it can often be automated through efficient verification algorithms and because it gives the programmer a clear comprehension of security requirements and mechanisms. We will consider both high (i.e., application) level properties as, e.g., information flow and "Service-Oriented" security, and low (i.e., communication) level properties as, e.g., authentication, secrecy and non-repudiation on standard and ad-hoc networks. We will also study how results achieved on "standard" symbolic models scale to computational and causal models, the former providing a more concrete representation of cryptography and the latter expressing security properties in terms of explicit cause-and-effect relations.

As illustrated in more detail in the following sections), our job is of a foundational nature, since it focuses on the definition and development of formal methodologies for the analysis of various aspects of information security.

2 National and International background

This job will focus on diverse research topics related to the application of formal methods to security, that we shortly describe below.

2.1 Communication and Network Security

Cryptographic protocols are one of the fundamental mechanisms for achieving security on computer networks. Wide-area networks are, in fact, not controllable and there is a need to protect sent/received data through cryptographic techniques. Even if these protocols are often just a few lines of codes, many attacks subverting the protocol logic and invalidating the expected security properties have been found. These attacks are not necessarily based on cryptographic flaws and can be reproduced even when cryptography is considered as a fully reliable black box. In the literature we find a huge amount of contributions on the analysis and verification of security protocols, but only a few of them

follow a language-based approach, i.e., are based on static-analysis. We mention here some relevant papers on secrecy [A99,AB05] and authentication [BBDNN05, BFM07, GJ03, GJ04]. We intend to go on on this line of research by focussing on abstract interpretation and control flow analysis of cryptographic protocols and abstract communication primitives to make programming independent of cryptographic implementation. Moreover, we also aim to extend the symbolic protocol analysis approach [AVISPA,Bla01,RSGLR00] in order to allow for the specification and verification of a larger class of protocols and properties than currently possible, as well as of different attackers models, extending preliminary work such as [HDMV05,HDMVB06].

We will finally consider security on Ad-hoc networks. A Mobile Ad-Hoc Network is an autonomous system composed of devices communicating with each other via radio transceivers. Mobile devices are free to move randomly and organize themselves arbitrarily; thus, the network's wireless topology may change rapidly and unpredictably. Trust establishment in the context of ad-hoc networks is still an open and challenging field [Gli04,PM06], because of lack of a fixed networking infrastructure, high mobility of the devices, limited range of the transmission, shared wireless medium, and physical vulnerability. We would like to develop formal models of trust that fit the constrains of ad-hoc networking, integrating these models in a process calculus for ad-hoc networks [NH06,Mer07,God07], thus developing an appropriate theory to formally prove security properties.

2.2 Application Security

Controlling information flow in programs and systems is a fundamental security issue whose theoretical foundations have been extensively studied. The aim is to control secret information so that it cannot flow towards unprivileged users who do not have the clearance to access it. Non-Interference is one of the reference properties for achieving this kind of control, and it was introduced by Goguen and Meseguer in [GM82]. The main idea is to require that any possible modifications of high level data have no observable effects at lower levels or, in other words, do no interfere with lower views of the system. In literature, we find many variants and extensions of Non-Interference on process calculi and simple imperative languages; see, e.g., [BCF02, BCFLP04, FG95, FRS05, GM04, MSZ06, RWW96, RS01, SS00, SM03, SV98]. Our research will specifically focus on Information Flow Security of distributed programs with cryptography, secure refinement of programs, security of (a multi-threaded fragment of) Java and extending the abstract non-interference framework [GM04] in order to deal with more powerful attackers.

Security also plays a crucial role in Service Oriented Computing. In this scenario, applications are built by assembling stand-alone components distributed over a network, called services. Services are open, i.e., built with little or no knowledge about their operating environment, their clients, and further services. Therefore, their secure composition and coordination may require peculiar mechanisms. Web Services [S02] built upon XML technologies are probably the most illustrative and well developed example of this paradigm. We intend to extend the results of [BDF06a, BDF06b], where we propose an approach based on semantic descriptions and a methodology which automates the process of discovering services and planning their composition in a secure way. Moreover, we plan to scale up the techniques developed for protocol analysis to security services. There are a number of preliminary approaches in this direction [BMPV06,SAMOA], but none of them has yet reached the required maturity.

2.3 Quantitative Aspects of Security

There are recent papers studying how formal analysis scales to computational security, a model of security requiring resistance over all the possible probabilistic polynomial-time attacks. This model, differently from Dolev-Yao, does not consider cryptography as a secure black box (see, e.g., [AR00, BCK05, BPW03, L05]). A formal symbolic analysis, à la Dolev-Yao, is typically simpler and easier to automate with respect to computational models. It is thus appealing to understand how symbolic formal results scale to these models and under which cryptographic assumptions this may happen. Even in this setting, the language-based approach has not been extensively studied. An interesting paper in this direction is [L05], which proposes a type system for message secrecy. It exploits a semantics based on the "simulatable cryptographic library" [BPW03] to scale the results to computational models. We intend to develop a static analysis based on [BFM07] for the verification of authentication protocols using the "simulatable cryptographic library".

We also intend to explore hybrid models that combine the two approaches: symbolic and computational. There is already a related literature [PW01, CCKLLPS06, MRST06,CP07] that in particular highlights a fundamental role of nondeterminism, for which an arbitrary resolution may lead to undesired conclusions. Thus, the main open problems are a correct management of nondeterminism and the study of hierarchical techniques that take care of computational aspects as well. The recent case study [ST07] analyses a simple and well known authentication protocol using Probabilistic Automata and a new notion of computationally bounded approximated simulation that allows an abstract system to emulate computational steps of a concrete system up to some negligible error. This case study constitutes a significant starting point for developing hierarchical and compositional proof methods for security..

2.4 Causal models for security

In the literature on cryptographic protocols analysis, we find some recent approaches in which the causal dependencies among events play a very important role [BCM07,CW01,FHG98,P99]. Strand spaces [FHG98] are a well known method in which causal dependencies are made explicit. In the inductive method of [P99], dependencies are instead a consequence of inductive rules. Proved Transition systems [DP92,DP99] represent an extension of transition systems towards causality. Proved Transition Systems can be considered as a sort of compact representation of computations, containing all the possible encodable and relevant information. Transitions are enriched with labels encoding their proofs, i.e. the steps involved in the deduction process of the action just executed. By inspecting the transition labels, it is possible to infer the causal dependencies, represented through a set of references to previous transitions. Starting from the enhanced semantics of [BetAl05], we intend to investigate the possible application of causal semantics based on Proved Transition Systems and on true-concurrent models, to the analysis of cryptographic protocols.

The Distributed State Temporal Logic (DSTL) [MSS04] permits to causally relate properties, which might hold in distinguished components of a system, in an asynchronous setting. The logic includes a primitive operator to specify events, thus allowing us to mix conditions and events in the specification formulae. The ability to deal with events explicitly enhances the expressiveness and simplicity of logical specifications, and seems especially adequate in the case of security properties specification. Starting from [MS04], we intend to further investigate the use of DSTL for the specification and verification of applications in which components presents various security requirements.

3 Results and Suggestions

Information security is becoming more and more relevant given the increasing usage of computers and networks for critical applications as, e.g, e-commerce, home-banking, purchase of digital goods and, in general, on-line services. It becomes thus very relevant to understand in depth the security requirements of distributed applications and to investigate methods for the automated verification of such requirements. The primary aim of the job is the study of foundations of information security and the development of formal methods for the specification and verification of security properties of programs, systems and computer networks.

We intended to cover many different aspects of security working both on high (i.e., application) level properties as, e.g., information flow and "Service-Oriented" security, and on low (i.e., communication) level properties as, e.g., authentication, secrecy and non-repudiation on standard and ad-hoc networks. Regarding formal methods, we mainly intended to investigate "language-based" techniques, which have the advantage of verifying security of programs directly on the code, without the need of analysing their execution. We believe that this approach was particularly appealing both because it could often be automated through efficient verification algorithms and because it gave the programmer a clear comprehension of security requirements and mechanisms.

We divided the work in four that reflect the logical and temporal scheduling of activities, corresponding to a "standard agenda" of the development of formal methods for security:

Step 1 - Security oriented languages and models. We studied security oriented languages, i.e., languages specifically developed for the specification and verification of security properties.

Step 2 - Security properties. We studied and formalized security properties on the languages defined in the previous step

Step 3 - Analysis techniques. We studied analysis techniques for the properties and languages described above. We implemented and extended verification tools based on the above mentioned techniques

3.1 Communication and Network Security.

We are interested in the analysis of cryptographic protocols through abstract interpretation, type systems, control flow analysis and causal semantics . We planned to extend the study of cryptographic protocols to distributed applications based on cryptography, by integrating this study with the program analysis techniques. We proposed new security-oriented languages and process calculi for distributed systems. We developed a logic for expressing local and global properties of distributed systems. Finally, we studied security models for ad-hoc networks.

3.2 Application Security.

We studied different aspects of program security through abstract interpretation: in particular, we are interested in models and methods for verifying non-interference in presence of active attackers and in probabilistic computations; we have dealtl with confidentiality and, in particular, both with "termination covert channels" in which the attacker gets information by observing the program termination, and with "timing covert channels". We studied properties for the secure refinement of programs in order to achieve a step-by-step development of secure applications, starting from abstract specifications. Finally, we studied primitives for the secure composition of clients and services in the setting of "Service-Oriented Computing".

3.3 Quantitative Aspects of Security.

We intended to study how properties described above, scale on finer-grained models, in which time and probabilities are explicitly modeled. We studied techniques to detect and remove timing attacks, by transforming a program so that its timing behavior is corrected while the input/output behavior is preserved. We also intended to develop new analysis techniques for computational security of cryptographic protocols. On the one hand, we developed type-based techniques for the correctness of protocols expressed on the cryptographic library proposed by Backes-Pfitzmann-Waidner; on the other hand, we studied soundness results of the symbolic model with respect to the computational model, through the work on approximated simulation relations of Segala and Turrini.

3.4 Causal models for security.

In the formalization of security properties it might be beneficial to reason in terms of causality among events. For example, in entity authentication we have that authentication should always be caused by the actual execution of the protocol by the claimant. We intended to give a new causal semantics to cryptographic protocols which enables us to directly observe the causality between the protocol conclusion, i.e., the authentication, and the corresponding execution by the authenticated entity. In doing this, we investigated both true-concurrent models like, e.g., event structures, and models of causality based on proved transition systems.

4 Conclusions

For each part of the job we give a list of the main results. These results are intentionally very specific, so to be verifiable.

For the Communication and Network Security:

- new formal models of trust for ad-hoc networks and integration of these models into suitable process calculi.
- a new security-oriented temporal logic for communicating processes.
- definition of an abstract interpretation of challenge-response authentication protocols;
- definition of new Control Flow Analyses for security protocols;
- extension of the verification tool proposed in [BBDNN05] to the new Control Flow Analyses;
- use of symbolic techniques and refinement for the verification of security properties;
- extension of AVISPA to the logic described;
- investigation of possible extensions of AVISPA to other techniques developed.

For the Application Security :

- definition of security-oriented imperative languages with cryptographic communication primitives;
- new abstract communication primitives that make programming independent of cryptographic implementation.
- new general framework for secure stepwise refinement of programs;
- new dynamic type systems for the security of distributed applications with cryptography;
- extension of the call-by-property invocation mechanism of Service-Oriented Computing to other security properties and non-functional aspects;
- extension of existing orchestration techniques to scenarios in which services may be published on-the-fly and may become temporarily unavailable;
- extension of abstract Non-Interference in order to deal with active attackers able to exploit probabilistic techniques;
- application of abstract Non-Interference to data bases and data mining.

For the Quantitative Aspects of Security :

- a new calculus for cryptographic protocols, with both a symbolic and a computational semantics based on the simulatable cryptographic library by Backes, Pfitzmann and Waidner [BPW03];
- a new hybrid model for security protocols combining symbolic and computational aspects;
- an extension of the process calculus LySa which is able to deal with type misinterpretation attacks.
- type systems for cryptographic protocols with both a symbolic and a computational semantics.

For the Causal models for security :

- new causal semantics for existing calculi of cryptographic protocols.
- new causality-based formalizations of security properties;
- new formalizations of security properties using the logic DSTL.
- specializations of already studied techniques to the new semantics, with special attention to authentication protocols

5 References

[A99] M. Abadi. Secrecy by Typing in Security Protocols. Journal of the ACM, 46(5):749–786, 1999.

[AB05] M. Abadi and B. Blanchet. Analyzing Security Protocols with Secrecy Types and Logic Programs. Journal of the ACM, 52(1):102–146, 2005

[AR00] M. Abadi and P. Rogaway. Reconciling Two Views of Cryptography (The Computational Soundness of Formal Encryption). In proc. of IFIP TCS 2000 (LNCS 1872) pp. 3–22.

[AVISPA] The AVISPA Project. www.avispa-project.org

[BBDNN05] C.Bodei, M.Buchholtz, P.Degano, F.Nielson, H.R.Nielson. Static Validation of Security Protocols. JCS 13(3), 2005.

[BCF02] C.Braghin, A. Cortesi, and R. Focardi. Security Boundaries in Mobile Ambients. Computer Languages, 28(1):101-127, 2002

[BCFLP04] C. Braghin, A. Cortesi, R. Focardi, F.L. Luccio, and C. Piazza. Nesting Analysis of Mobile Ambients. Computer Languages, Systems & Structures 30(3-4):207-230, 2004

[BCK05] M. Baudet, V. Cortier and S. Kremer. Computationally Sound Implementations of Equational Theories against Passive Adversaries, In Proc. of ICALP'05. LNCS 3580. pp. 652-663.

[BCM07] M. Backes, A. Cortesi, M. Maffei. Abstracting Multiplicity in Cryptographic Protocols. In Proc. of IEEE CSF'07, pp. 355-369

[BDF06a] M.Bartoletti, P.Degano, G.L.Ferrari. Plans for service composition. Proc. of WITS, 2006.

[BDF06b] M.Bartoletti, P.Degano. G.L.Ferrari. Types and effects for secure orchestration. Proc. of IEEE CSFW, 2006.

[BetAl5] C.Bodei, M.Buchholtz, P.Degano, M.Curti, C.Priami, F.Nielson, H.R.Nielson. On Evaluating the Performance of Security Protocols specified in Lysa. Proc. of PACT05, LNCS 3606. Ree Source Person. “Title of Research Paper”; name of journal (name of publisher of the journal), Vol. No., Issue No., Page numbers (eg.728—736), Month, and Year of publication (eg. Oct 2006).

[BFM07] M. Bugliesi, R. Focardi and M. Maffei. Dynamic Types for Authentication, Journal of Computer Security, IOS Press, 15(6):563-617, 2007

[Bla01] B. Blanchet. An efficient cryptographic protocol verifier based on prolog rules. IEEE CSFW’01

[BMPV06] M. Backes, S. Moedersheim, B. Pfitzmann, L. Vigano`. Symbolic and Cryptographic Analysis of the Secure WS-Reliable Messaging Scenario. In Proc. of FOSSACS’06. LNCS 3921.

[BPW03] M. Backes, B. Pfitzmann, and M. Waidner. A Universally Composable Cryptographic Library. In proc. of ACM CCS 2003, pp. 220-230.

[CCKLLPS06] R. Canetti, L. Cheung, D. Kirli Kaynar, M. Liskov, N. A. Lynch, O. Pereira, R. Segala: Time-Bounded Task-PIOAs: A Framework for Analyzing Security Protocols. In Proc. of DISC’06. LNCS 4167, pp 238-253.

[CP07] K. Chatzikokolakis, C. Palamidessi. Making Random Choices Invisible to the Scheduler. In Proc. of CONCUR’07. LNCS 4703, pp. 42-58.

[CW01] F.Crazzolara, G.Winskel. Events in Security Protocols. In ACM CCS, 2001.

[DP92] P.Degano, C.Priami. Proved Trees. In Proc. of ICALP'92.

[DP99] P.Degano, C.Priami. Non Interleaving Semantics for Mobile Processes. TCS 216, 1999.

[FG95] R. Focardi and R. Gorrieri, A Classification of Security Properties for Process Algebras, Journal of Computer Security, 3(1):5-33, 1995

[FHG98] F.J.T. Fábrega, J.C.Herzog, J.D.Guttman. Strand spaces: Why is a security protocol correct? JCS 7(2-3), 1999.

[FRS05] R. Focardi, S. Rossi, A. Sabelfeld: Bridging Language-Based and Process Calculi Security. In Proc. of FoSSaCS 2005 pp. 299-315. LNCS 3441

[GJ04] A. D. Gordon and A. Jeffrey. Types and effects for asymmetric cryptographic protocols. Journal of Computer Security, 12(3-4):435–483, 2004

[Gli04] V.D.Gligor. Security of Emergent Properties in Ad-Hoc Networks. Security Protocols Workshop 2004

[GM04] R. Giacobazzi and I. Mastroeni. Abstract Non-Interference. POPL'04

[God07] J.C. Godskesen. A Calculus for Mobile Ad Hoc Networks. COORDINATION'07

[HDMV05] P. Hankes Drielsma, S. Moedersheim, L. Vigano`. A Formalization of Off-Line Guessing for Security Protocol Analysis. LPAR04

[HDMVB06] P. Hankes Drielsma, S. Moedersheim, L. Vigano`, D. Basin. Formalizing and Analyzing Sender Invariance. FAST'06

[L05] P. Laud. Secrecy types for a simulatable cryptographic library. In Proc. of the 12th ACM CCS '05. New York, NY, 26-35.

[MRST06] J. C. Mitchell, A. Ramanathan, A. Scedrov, V. Teague. A probabilistic polynomial-time process calculus for the analysis of cryptographic protocols. TCS 353, 2006

[MSZ06] A. C. Myers, A. Sabelfeld, and S. Zdancewic. Enforcing Robust Declassification. Journal of Computer Security, 14(2):157-196, 2006.

[NH06] S.Nanz, C.Hankin. A Framework for Security Analysis of Mobile Wireless Networks. TCS 367, 2006

[Mer07] M.Merro. On the Observational Theory of Mobile Ad-Hoc Networks. Information and Computation, to appear.

[MSS04] C.Montangero, L. Semini and S. Semprini. Logic Based Coordination for Event-Driven Self-Healing Distributed Systems. Proc. of COORDINATION'04, LNCS 2949.

[MS04] C.Montangero, L. Semini. Formalizing an Adaptive Security Infrastructure in Mobadtl. Proc. of FCS'04.

[P99] L.C.Paulson. Proving security protocol correct. Proc. of Lics, 1999.

[PM06] A.A. Pirzada, C.McDonald. Trust Establishment in Pure Ad-Hoc Networks. Wireless Personal Communication 379, 2006

[PW01] B. Pfitzmann, M. Waidner: A Model for Asynchronous Reactive Systems and its Application to Secure Message Transmission. IEEE Symposium on S&P 2001

[RS01] P. Ryan and S. Schneider, Process algebra and Non-Interference, Journal of Computer Security 9(1/2):75–103, 2001.

[RSGLR00] P. Ryan, S. Schneider, M. Goldsmith, G. Lowe, and B. Roscoe. Modelling and Analysis of Security Protocols. 2000

[RWW96] A.W. Roscoe, J.C.P. Woodcock and L. Wulf, Non-Interference through determinism, Journal of Computer Security 4(1):27-54, 1996.

[S02] M.Stal. Web services: Beyond component-based computing. Comms. Of the ACM 55(10), 2002.

[SAMOA] Samoa: Formal Tools for Securing Web Services. http://research.microsoft.com/projects/samoa/.

[SS00] A. Sabelfed and D. Sands, Probabilistic Noninterference for multi-threaded programs, in: Proc. of IEEE CSFW 2000, pp.200-215.

[SM03] A. Sabelfeld and A.C. Myers, Language-based information-flow security, IEEE Journal on Selected Areas in Communication 21(1):5-19, 2003.

[ST07] R. Segala, A. Turrini. Approximated Computationally Bounded Simulation Relations for Probabilistic Automata. IEEE CSF07

[SV98] G. Smith and D.M. Volpano, Secure information flow in a multi-threaded imperative language, in: Proc. of 25th ACM POPL, pp.355-364, 1998