INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team Cassis

Combining approaches for the security of infinite state systems

Nancy - Grand Est

Theme : Programs, Verification and Proofs

Activity Report

2009

Table of contents

1. Team

Research Scientist

Serge Burckel [MC, U. de la Réunion, seconded to INRIA, HdR]
Yannick Chevalier [MC, U. Paul Sabatier, Toulouse, seconded to INRIA]
Véronique Cortier [CR, CNRS-LORIA, HdR]
Christophe Ringeissen [CR, INRIA-LORIA, HdR]
Michaël Rusinowitch [Team Leader, Research Director (DR), INRIA-LORIA, HdR]
Mathieu Turuani [CR, INRIA-LORIA]
Silvio Ranise [CR, INRIA-LORIA, in sabbatical stay, Univ. of Milan]

Faculty Member

Fabrice Bouquet [PR, Université Franche-Comté, HdR]
Frédéric Dadeau [MC, Université Franche-Comté]
Alain Giorgetti [MC, Université Franche-Comté]
Pierre-Cyrille Héam [MC, Université Franche-Comté, seconded to CNRS, Cachan until August 31, HdR]
Olga Kouchnarenko [Vice-head of project team, PR, Université Franche-Comté, LIFC, HdR]
Abdessamad Imine [MC, Université Nancy 2]
Laurent Vigneron [MC, Université Nancy 2]

Technical Staff

Aloïs Dreyfus [Engineer ANR RAVAJ, LIFC]
Kalou Cabrera [Engineer ODL, LIFC]
Philippe Paquelier [Engineer FP7 SecureChange, LIFC, from February 1]

PhD Student

Mumtaz Ahmad [SFERE (Pakistan), LORIA]
Mathilde Arnaud [project AVOTÉ, from September 1]
Tigran Avanesov [INRIA, LORIA]
Asma Berregba [MENRT, LORIA]
Thibaut Brocard [BDI-CNRS, LIFC until September 30 and ATER UFC]
Pierre-Christophe Bué [MENRT, LIFC, from October 1]
Najah Chridi [MENRT, LORIA, thesis defended on September 11]
Stefan Ciobaca [project AVOTÉ]
Roméo Courbis [LIFC]
Stéphane Debricon [INTERREG, LIFC]
Elizabeta Fourneret [FP7 SecureChange, LIFC, from September 1]
Adrien de Kermadec [VALMI, Co-tutelle LIFC and New-Zeland]
Jonathan Lasalle [project VETESS, LIFC]
Mohamed Anis Mekki [INRIA, LORIA]
Vincent Pretre [INTERREG, LIFC, thesis defended on March 18]
Elena Tushkanova [INRIA, LIFC, from November 1]

Post-Doctoral Fellow

Enrica Nicolini [Post-doctoral INRIA, until November 30, 2009]

Administrative Assistant

Emmanuelle Deschamps

2. Overall Objectives

2.1. Background

Cassis is a joint project between the *Laboratoire Lorrain de Recherche en Informatique et ses Applications (LORIA - UMR 7503)* and *Laboratoire d'Informatique de l'Université de Franche-Comté (LIFC - FRE 2661).*

The objective of the project is to design and develop tools to verify the safety of systems with an infinite number of states. The analysis of such systems is based on a symbolic representation of sets of states in terms of formal languages or logical formulas. Safety is obtained via automatic proof, symbolic exploration of models or test generation. These validation methods are complementary. They rely on the study of accessibility problems and their reduction to constraint solving.

An originality of the project is its focus on infinite systems, parameterized or large scale, for which each technique taken separately shows its limits. This is the case for example with protocols operating on topologies of arbitrary size (ring networks), systems handling data structures of any size (sets), or whose control is infinite (automata communicating through an unbounded buffer). Ongoing or envisioned applications concern embedded software (e.g., smart cards, automotive controllers), cryptographic protocols (IKE, SET, TLS, Kerberos) designed to ensure trust in electronic transactions, and distributed systems.

The problem of validating or verifying reactive systems is crucial because of the increasing number of security-sensitive systems. The failure of these critical systems can have dramatic consequences since they may be embedded in vehicles components, or they control power stations or telecommunication networks. Beside obvious security issues, the reliability of products whose destination is millions of end-users has a tremendous economical impact.

There are several approaches to system verification: automated deduction, reachability analysis or model-checking, and testing. These approaches have different advantages and drawbacks. Automated deduction can address practical verification, however it remains complex to handle and requires a lot of expertise and guidance from the user. Model-checking is exhaustive but must face combinatorial explosion and becomes problematic with large-size or infinite systems. Testing is fundamental for validating requirements since it allows the discovery of many errors. However, it is almost never exhaustive and therefore only leads to partial solutions. Hence we believe that these approaches should not be considered as competing but as complementary.

The goal of our project is to contribute to new combinations of these three verification techniques in a framework that would apply them in an industrial context. In particular we expect some breakthrough in the infinite-state verification domain by joint applications of deductive, model-checking and testing techniques.

2.2. Context

For verifying the security of infinite state systems we rely on

- Different ways to express the safety, reachability or liveness properties of systems, linear-time or branching-time logics, and the application of abstraction or abstract interpretation.
- Test generation techniques.
- The modeling of systems by encoding states as words, terms or trees and by representing infinite sets of states by languages. To each of these structures corresponds appropriate action families, such as transductions or rewritings.

Our goal is to apply these different approaches for ensuring the security of industrial systems by providing adequate methods and tools. In more details we aim at the following contributions (see the continuous lines in Figure 1):

1. verification of abstract models derived from existing systems;
2. tests generation from the abstract model for validating the existing model;
3. cross-fertilization of the different validation techniques (deduction, model-checking, testing) by taking advantage of the complementary scopes and of their respective algorithmic contributions.

Let us mention that all these techniques comply with various development methodologies.

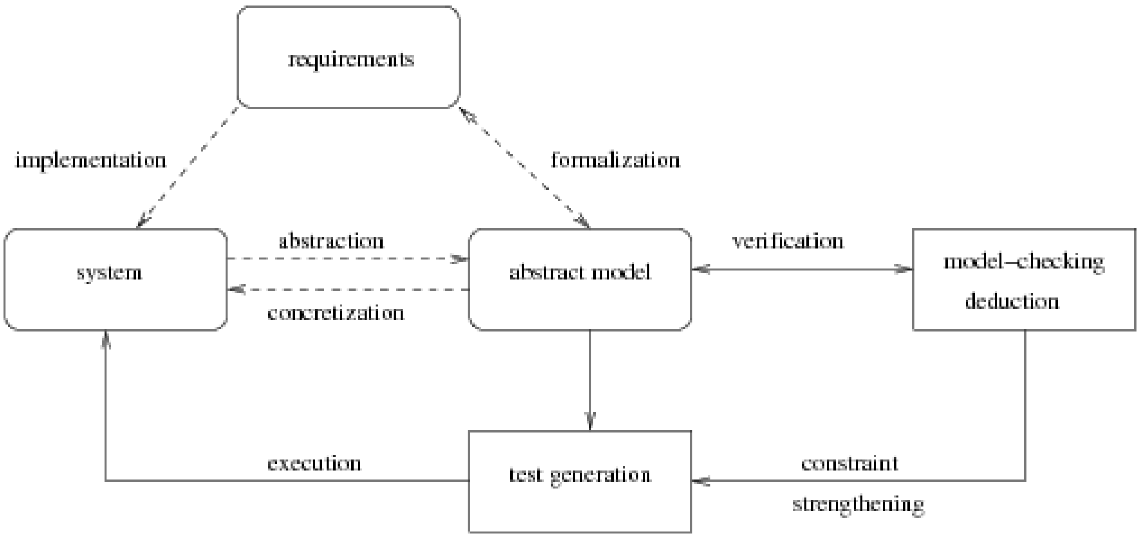

Figure 1. Software validation in Cassis

2.3. Challenge

Verifying the safety of infinite state systems is a challenge: nowadays algorithmic techniques only apply to very specific infinite state systems. On the other hand the deductive approaches are good candidates to capture infinite system safety verification but are difficult to bring into operation and require a deep expertise. A solution consists of integrating several verification methods by combining, for example, theorem-proving and model-checking.

The behavior of infinite states systems is expressed in the various models by composing or iterating actions. One of the main problems with algorithmic techniques is to compute the effect of these actions on the initial state. This computation is called *reachability analysis*. The verification of safety properties as well as the automatic generation of test cases relies heavily on the accuracy of reachability analysis.

The transverse goal is to push away the limitations on the use of formal verification techniques, to ease their applications, and to let them scale-up.

1. For properties that can be checked by reachability analysis we have proposed models based on regular languages and rational transductions. We have completed them by designing algorithms for verifying a refinement relation between two models $\mathcal{S}$ and $\mathcal{T}$ [63]. This refinement relation when satisfied preserves the safety properties and therefore allows them to be inherited. We shall investigate this approach with other representations.
2. In order to generate boundary-value functional test cases, we abstract models as constrained states. These constraints are solved by a customized solver, called CLPS. The test cases are derived in two steps [6]:
   1. partitioning of the formal model and extraction of boundary values,
   2. reachability graph exploration from constrained states in order to reach boundary values and generate state sequences (traces) as test cases with the oracle.

   After the generation phase, a concretization is used to produce the test drivers [7]. Furthermore, the kernel of the engine allows one to perform specification animations in order to validate the model [69].

3. For the safety of infinite state systems we have designed automated deduction tools based on term rewriting (*SPIKE*, *daTac*, *haRVey*) and an extensible and modular platform for detecting flaws and potential attacks on security protocols (*AVISPA*). The tools have been built on the modeling of systems by terms and rewrite rules. Our work with other models based on regular languages of words or trees and of transducers should complement these term rewriting models.

In order to address this challenge, we rely on complementary skills within the project. We believe that each of the three techniques will benefit from concepts and algorithms designed for the two others.

2.4. Highlights

Three members of the EPI Cassis have defended their Habilitation: Véronique Cortier, Pierre-Cyrille Héam and Christophe Ringeissen.

3. Scientific Foundations

3.1. Introduction

Our main goal is to design techniques and to develop tools for the verification of (safety-critical) systems, such as programs or protocols. To this end, we develop a combination of techniques based on automated deduction for program verification, constraint resolution for test generation, and reachability analysis for the verification of infinite state systems.

3.2. Automated Deduction

The main goal is to prove the validity of assertions obtained from program analysis. To this end, we develop techniques and automated deduction systems based on rewriting and constraint solving. The verification of recursive data structures relies on inductive reasoning or the manipulation of equations and it also exploits some form of reasoning modulo properties of selected operators (such as associativity and/or commutativity).

Rewriting, which allows us to simplify expressions and formulae, is a key ingredient for the effectiveness of many state-of-the-art automated reasoning systems. Furthermore, a well-founded rewriting relation can be also exploited to implement reasoning by induction. This observation forms the basis of our approach to inductive reasoning, with high degree of automation and the possibility to refute false conjectures.

The constraints are the key ingredient to postpone the activity of solving complex symbolic problems until it is really necessary. They also allow us to increase the expressivity of the specification language and to refine theorem-proving strategies. As an example of this, the handling of constraints for unification problems or for the orientation of equalities in the presence of interpreted operators (e.g., commutativity and/or associativity function symbols) will possibly yield shorter automated proofs.

Finally, decision procedures are being considered as a key ingredient for the successful application of automated reasoning systems to verification problems. A decision procedure is an algorithm capable of efficiently deciding whether formulae from certain theories (such as Presburger arithmetic, lists, arrays, and their combination) are valid or not. We develop techniques to build and combine decision procedures for the domains which are relevant to verification problems. We also perform experimental evaluation of the proposed techniques by combining propositional reasoning (implemented by means of Boolean solvers – Binary Decision Diagrams or SAT solvers) and decision procedures, and their extensions to semi-decision procedures for handling larger (possibly undecidable) fragments of first-order logic.

We investigate techniques to incorporate the use of decision procedures in the model-checking of infinite state systems. The state of such systems is described by the models of theories specifying data types (such as integers or arrays) and their behavior is identified by (possibly infinite) sequences of these models which share the interpretation of the symbols interpreted in the theories (e.g., the addition over the integers). In this context, checking if a system satisfies a certain property may be reduced to checking the satisfiability of a formula in the theory obtained as the combination of the theories describing the sequence of states in the computation. To solve this problem, it is crucial to develop new combination methods for non-disjoint unions of theories.

3.3. Synthesizing and Solving Set Constraints

Applying constraint logic programming technology in the validation and verification area is currently an active way of research. It usually requires the design of specific solvers to deal with the description language's vocabulary. We are interested in using a solver for set constraints based on the CLPS core [2], to evaluate set-oriented formal specifications. By evaluation, we mean the encoding of the formal model into a constraint system, and the ability for the solver to verify the invariant on the current constraint graph, to propagate preconditions or guards, and to apply the substitution calculus on this graph. The constraint solver is used for animating specifications and automatically generating abstract test cases.

3.4. Rewriting-based Safety Checking

Invariant checking and strenghtening is the dual of reachability analysis, and can thus be used for verifying safety properties of infinite-state systems. In fact, many infinite-state systems are just parameterized systems which become finite state systems when parameters are instantiated. Then, the challenge is to automatically discharge the maximal number of proof obligations coming from the decomposition of the invariance conditions. For parameterized systems, we develop a deductive approach where states are defined by first order formulae with equality, and proof obligations are checked by the automatic theorem prover *haRVey*. Thanks to this tool, we study the applicability of the superposition calculus (a modern version of resolution with a built-in treatment of the equality predicate and powerful techniques for reducing the search space) for deciding conditions arising from program verification.

4. Application Domains

4.1. Verification of Security Protocols

Security protocols such as SET, TLS and Kerberos, are designed for establishing the confidence of electronic transactions. They rely on cryptographic primitives, the purpose of which is to ensure integrity of data, authentication or anonymity of participants, confidentiality of transactions, etc.

Experience has shown that the design of those protocols is often erroneous, even when assuming that cryptographic primitives are perfect, i.e., that an encoded message cannot be decrypted without the appropriate key. An intruder can intercept, analyze and modify the exchanged messages with very few computations and therefore, for example, generate important economic damage.

Analyzing cryptographic protocols is complex because the set of configurations to consider is very large, and can even be *infinite*: one has to consider any number of sessions, any size of messages, sessions interleaving, some algebraic properties of encryption or data structures.

Our objective is to automatize as much as possible the analysis of protocols starting from their specification. This consists in designing a tool easy to use, permitting to specify a large number of protocols thanks to a standard high-level language, and permitting either to look for flaws in a given protocol or to check whether it satisfies a given property. Such a tool is essential for verifying existing protocols, but also for helping in designing new ones. For our tool to be easy to use, it has to provide a graphical interface allowing a user to do only click-button.

Our tools for verifying security protocols are available as components of the AVISPA platform. As an extension of the AVISPA specification language, we are working on a new environment called *CASRUL* for handling more general protocols like e-business protocols for example.

4.2. Automated Boundary Testing from Formal Specifications

In [7], we have presented a new approach for test generation from set-oriented formal specifications: the BZ-TT method. This method is based on Constraint Logic Programming (CLP) techniques. The goal is to test every operation of the system at every boundary state using all input boundary values of that operation. It has been validated in several industry case studies for smart card OS and application validation (GSM 11-11 standard [64] and Java Card Virtual Machine Transaction mechanism [68]) and for embedded automotive software (an automobile wind-screen wiper controller).

This test generation method can be summed up as follows: from the formal model, the system computes boundary values to create boundary states; test cases are generated by traversal of the state space with a preamble part (sequences of operations from the initial state to a boundary state), a body part (critical invocations), an identification part (observation and Oracle state computation) and a post-amble part (return path to initial or boundary state). Then, an executable test script file is generated using a test pattern and a table of correspondence between abstract operations (from the model) and concrete ones. This approach differs in several main points from the work of Dick, Faivre *et al*: first, using boundary goals as test objectives avoids the complete construction of the reachability graph; second, this process is fully automated and the test engineer could just drive it at the boundary value computation level or for the path computation.

The BZ-TT method is fully supported by the BZ-Testing-Tools tool-set. This environment is a set of tools dedicated to animation and test cases generation from B, Z or State-Chart formal specifications. It is based on the CLPS constraint solver, able to simulate the execution of the specification. By execution, we mean that the solver computes a so-called constrained state by applying the pre- and post-condition of operations. A constrained state is a constraint store where state variables and also input and output variables support constraints.

One orientation of the current work is to go beyond the finiteness assumption limitations by using symbolic constraint propagation during the test generation process and to extend the result to object oriented specifications.

4.3. Program Debugging and Verification

Catching bugs in programs is difficult and time-consuming. The effort of debugging and proving correct even small units of code can surpass the effort of programming. Bugs inserted while “programming in the small” can have dramatic consequences for the consistency of a whole software system as shown, e.g., by viruses which can spread by exploiting buffer overflows, a bug which typically arises while coding a small portion of code. To detect this kind of errors, many verification techniques have been put forward such as static analysis and software model checking.

Recently, in the program verification community, there seems to be a growing demand for more declarative approaches in order to make the results of the analysis readily available to the end user. To meet this requirement, a growing number of program verification tools integrate some form of theorem proving.

The goals of our research are twofold. First, we perform theoretical investigations of various combinations of propositional and first-order satisfiability checking in order to automate the theorem proving activity required to solve a large class of program analysis problems which can be encoded as first-order formulae. Second, we experimentally investigate how our techniques behave on real problems so to make program analysis more precise and scalable. Building tools capable of providing a good balance between precision and scalability is one of the crucial challenges to transfer theorem proving technology to the industrial domains.

4.4. Towards New Application Domains

4.4.1. Web Services

Driven by rapidly changing requirements and business needs, IT systems and applications are undergoing a paradigm shift: components are replaced by services, distributed over the network, and composed and reconfigured dynamically in a demand-driven way into service-oriented architectures [1]. Exposing services in future network infrastructures means a wide range of trust and security issues need to be adressed. Solving them is extremely hard since making the service components trustworthy is not sufficient: composing services leads to new subtle and dangerous vulnerabilities due to interference between component services and policies, the shared communication layer, and application functionality. Thus, one needs validation of both the service components and their composition into secure service architectures. In this context, there is an obvious need of applying formal methods. Our project aims at applying our proof and constraint solving techniques to reason on web services. More precisely, we plan to focus on the composition problem in the presence of security policies.

4.4.2. Microrobotics

Researchers in microrobotics have recently proposed the concept of a distributed and integrated micromanipulator called *smart surface*, based on an array of smart micromodules in order to realize an automated positioning and conveying surface. Each micro-module will be composed of a micro-actuator, a micro-sensor and a control unit. The cooperation of these micromodules will allow to recognize the parts and to control micro-actuators on order to move and position accurately the parts on the smart surface.

Our objective is to elaborate new specification languages and verification methods to validate distributed smart surfaces at different levels of abstraction. We will bring our experience in formal verification, more especially in regular model-checking (RMC).

We collaborate with the AS2M (Automatique et Systèmes Micro-Mécatroniques) department at the FEMTO-ST (Franche-Comté Electronique Mecanique Thermique et Optique - Sciences et Technologies) institute (UMR 6174) on verifying and validating an adaptative *microfactory* model they have developed. We have defined a complete information model of multi-cells microfactories in UML. This model is used as the communication basis between the robotic and computing researchers. It includes the structure of the physical components of the microfactory - cells and transports functions - and the logical components - information gathering and exchange. The next step will be to provide properties and a dynamic model of microfactories.

5. Software

5.1. Protocols Verification Tools

Participants: Pierre-Cyrille Héam, Olga Kouchnarenko, Michaël Rusinowitch, Mathieu Turuani, Laurent Vigneron.

5.1.1. AVISPA

Cassis has been one of the 4 partners involved in the European project AVISPA, which has resulted in the distribution of a tool for automated verification of security protocols, named *AVISPA* Tool. It is freely available on the web [2] and supported. The *AVISPA* Tool compares favourably to related systems in scope, effectiveness, and performance, by (i) providing a modular and expressive formal language for specifying security protocols and properties, and (ii) integrating 4 back-ends that implement automatic analysis techniques ranging from *protocol falsification* (by finding an attack on the input protocol) to *abstraction-based verification* methods for both finite and infinite numbers of sessions.

In 2009, no new release of the AVISPA Tool has been delivered, but the users mailing-list has been active and an important contribution has been proposed by Thomas Genet (LANDE Project, IRISA), SPAN, a protocol animator.

The tool has also been used in the group for analyzing non-repudiation protocols.

[1] see e.g. http://osoa.org/display/Main/Service+Component+Architecture+Home

[2] http://www.avispa-project.org

5.1.2. CL-AtSe

We develop, as a first back-end of *AVISPA*, *CL-AtSe*, a Constraint Logic based Attack Searcher for cryptographic protocols. The *CL-AtSe* approach to verification consists in a symbolic state exploration of the protocol execution, for a bounded number of sessions. This necessary restriction (for decidability, see [72]) allows *CL-AtSe* to be correct and complete, i.e., any attack found by *CL-AtSe* is a valid attack, and if no attack is found, then the protocol is secure for the given number of sessions. Each protocol step is represented by a constraint on the protocol state. These constraints are checked lazily for satisfiability, where satisfiability means reachability of the protocol state. *CL-AtSe* includes a proper handling of sets (operations and tests), choice points, specification of any attack states through a language for expressing fairness, non-abuse freeness, etc..., advanced protocol simplifications and optimizations to reduce the problem complexity, and protocol analysis modulo the algebraic properties of cryptographic operators such as XOR (exclusive or) and Exp (modular exponentiation). The handling of XOR and Exp has required to implement an optimized version of the combination algorithm of Baader & Schulz [62] for solving unification problems in disjoint unions of arbitrary theories.

CL-AtSe has been successfully used by Cassis members [48] to analyse France Telecom R&D, Siemens AG, IETF, or Gemalto protocols in funded projects. It is also employed by external users, e.g., from the AVISPA's community. Moreover, *CL-AtSe* achieves very good analysis times, comparable and sometimes better than state-of-the art tools in the domain (see [76] for tool details and precise benchmarks).

5.1.3. TA4SP

We have developed, as a second back-end of *AVISPA*, TA4SP (Tree Automata based on Automatic Approximations for the Analysis of Security Protocols), an automata based tool dedicated to the validation of security protocols for an unbounded number of sessions. This tool provides automatic computations of over and under approximations of the knowledge accessible by an intruder. This knowledge is encoded as a regular tree language and protocol steps and intruder abilities are encoded as a term rewriting system. When given a reachability problem such as secrecy, TA4SP reports that (1) the protocol is safe if it manages to compute an over-approximation of intruder's knowledge that does not contain a secret term or (2) the protocol is unsafe in the rewrite model if it manages to compute an underapproximation of intruder's knowledge containing a secret term or (3) I don't know otherwise. TA4SP has verified 28 industrial protocols and case (3) occurred only once, for Kaochow protocol version 2.

TA4SP handles protocols using operators with algebraic properties. Thanks to a recent quadratic completion algorithm new experimental results have been obtained, for example for the Encrypted Key Exchange protocol (EKE2) using the exponential operator.

5.2. Testing Tools

Participants: Fabrice Bouquet, Frédéric Dadeau.

The Testing Tools is a tool-set for animation and test generation from B, JML, Z and State-chart specifications. It consists of two components:

- **BZ-Testing-Tools**– BZ-TT – is a tool-set for animation and test generation from B, Z and State-chart specifications. BZ-TT provides several testing strategies (partition analysis, cause-effect testing, boundary-value testing and domain testing), and several test model coverage criteria (multiple condition coverage, boundary coverage and transition coverage).

  A rebuild of the architecture of the BZ-Testing-Tools engine has started in December 2008, with the help of an "ingénieur jeune diplomé" from INRIA. It aimed at integrating the latest works on constraint solving and theorem proving, in a modular architecture dedicated to the analysis and exploitation of formal behavioral models for test generation purposes.
- **JML-Testing-Tools** – JML-TT – is a framework for the symbolic animation of formal models written using JML annotations [75] embedded within Java programs. JML-TT provides a simple and efficient way to semi-automatically validate a JML specification and to check model properties

such as class invariant or history constraints during the animation. This tool is used in the ACI GECCOO project[3].

We develop a third tool **Test-For-Testing-Tools** to validate the tests. The tool takes as input a code program and a test suite (realized by several approaches such as BZ-TT/random/properties driven tests). The system performs a mutation of the code program and we observe how many mutants are killed with each test suite.

5.3. Others Tools

Most of the software tools described in previous sections are using tools that we have developed in the past: BZ-TT uses the set constraints solver CLPS and *SPIKE*. Note that the development of the SMT prover haRVey has been stopped. The successor of haRVey in Nancy is called veriT and is developed by David Déharbe (UFRN Natal, Brasil) and Pascal Fontaine (MOSEL).

6. New Results

6.1. Automated Deduction

We develop general techniques which allow us to re-use available tools in order to build a new generation of satisfiability solvers offering a good trade-off between expressiveness, flexibility, and scalability. We focus on the careful integration of combination techniques and rewriting techniques to design satisfiability procedures for a wide range of (combined) theories of interest in verification.

6.1.1. Decision procedures for data structures combined with theories of arithmetic

Participants: Enrica Nicolini, Christophe Ringeissen, Michaël Rusinowitch.

We show how to use a non-disjoint extension of the Nelson-Oppen combination method to obtain decision procedures for theories modelling data structures and arithmetic constraints.

We propose a first solution when the incorporated arithmetic operator allows to express only linear increments, i.e. when the considered constraints have to be interpreted modulo the theory of integer offsets [44]. We present a superposition calculus dedicated to theories that model some data structures and that share the integer offsets; we show that the calculus is capable to actually decide the existential fragment of these theories and that can be plugged into the non-disjoint extension of the Nelson-Oppen combination method, deriving thus decision procedure for theories modeling more complex data structures.

As a second contribution [43], we focus on the union of a data-structure and a theory of arithmetic sharing a successor function satisfying the injectivity and the acyclicity axioms. This union allows us to handle more expressive arithmetic constraints and to obtain a combined decision procedure in which the procedures for individual theories can be constructed by using an appropriate superposition calculus for the data-structure and classical solving techniques for the theory of arithmetic (Gauss elimination, Fourier-Motzkin elimination, Groebner bases computation).

To go beyond a shared unary successor symbol, we consider the case of abelian groups [42]. The possibility of having a shared addition symbol permits us to augment the expressiveness on the arithmetical part, lifting from linear increment expressed by using the successor symbols, to increment expressed as sums. This allows to handle, e.g., useful counting functions for data structures such as trees. We consider the completeness and the effectiveness of the non-disjoint combination method when the theory of abelian groups is shared. For the completeness, we show that the theory of abelian groups can be embedded into a theory admitting quantifier elimination. For achieving effectiveness, we rely on a superposition calculus modulo abelian groups developed by Godoy and Nieuwenhuis. We consider a many-sorted and constraint-free version of the calculus, in which we use a restricted form of unification in abelian groups with free symbols, and in which only literals are involved.

[3] http://geccoo.lri.fr

To be effective in all our papers mentioned above, the non-disjoint extension of the Nelson-Oppen combination method makes use of procedures able to compute the logical consequences over the shared signature.

6.1.2. Hypothesis Selection

Participant: Alain Giorgetti.

Increasing the automaticity of proofs in deductive verification of C programs is a challenging task. When applied to industrial C programs known heuristics to generate simpler verification conditions are not efficient enough. This is mainly due to their size and a high number of irrelevant hypotheses. We [34] have presented a strategy to reduce program verification conditions by selecting their relevant hypotheses. The relevance of a hypothesis is determined by the combination of a syntactic analysis and two graph traversals. The first graph is labeled by constants and the second one by the predicates in the axioms. The approach is applied on a benchmark arising in industrial program verification.

6.1.3. Tree Automata and Rewriting

Participants: Michaël Rusinowitch, Laurent Vigneron.

In collaboration with F. Jacquemard (DAHU project) we pursue our investigation on rewriting systems for unranked ordered terms, i.e. trees where the number of successors of a node is not determined by its label, and is not a priori bounded. We model XML update operations with parametrized rewriting on unranked trees. Then we compute the forward and backward reachability sets of these systems for unranked trees languages given by several classes of hedge automata [58]. This gives more insight on these notions that have not been investigated before. In the context of XML processing, static type checking amounts verifying that a document transformation always converts valid source documents into valid output documents. We solve this problem for arbitrary sequences of atomic XML update operations from different subsets of the W3C XQuery Update Facility 1.0. We then apply the results to the verification of access control policies for XML updates. We propose an algorithm for the policy local consistency problem, that is, for deciding whether a sequence of authorized operations starting from a given document can simulate a forbidden one.

6.2. Security Protocol Verification

The design of cryptographic protocols is error-prone. Without a careful analysis, subtle flaws may be discovered several years after the publication of a protocol, yielding potential harmful attacks. In this context, formal methods have proved their interest for obtaining good security guarantees. Many analysis techniques have been proposed in the litterature [66]. We develop new techniques for richer primitives, wider classes of protocols and, higher security guarantees.

6.2.1. Modeling complex primitives

Participants: Véronique Cortier, Yannick Chevalier, Pierre-Cyrille Héam, Olga Kouchnarenko, Michaël Rusinowitch, Mathieu Turuani.

Some attacks exploit in a clever way the interaction between protocol rules and algebraic properties of cryptographic operators. In [71], we provide a list of such properties and attacks as well as existing formal approaches for analyzing cryptographic protocols under algebraic properties.

Focusing on ground deducibility and static equivalence (checking whether two sequences of messages are indistinguishable to an attacker), we have proposed [26] an efficient and generic decision procedure for a wide class of equational theories, including subterm convergent theories (e.g. encryption, signatures, pairing and hash) and layered convergent theories (e.g. blind signatures). The procedure is generic in the sense that it remains sound and complete (but may not terminate) for any convergent theory. It has been implemented in the YAPA tool[4]. We have also shown [53] that deducibility and static equivalence are decidable for the equational theories modeling trapdoor commitment and re-encryption, that are particularly relevant in the context of e-voting protocols.

[4] http://www.lsv.ens-cachan.fr/~baudet/yapa/

Encryption “distributing over pairs” is employed in several cryptographic protocols. As a first step towards solving intruder constraints under this hypothesis, we show that unification is decidable for an equational theory HE specifying such an encryption [23]. The method consists in transforming any given problem in such a way, that the resulting problem can be solved by combining a graph-based reasoning on its equations involving the homomorphisms, with a syntactic reasoning on its pairings.

We have also continued the work on the symbolic derivation model for cryptographic protocols that was introduced in [70]. We were in particular interested by the problem of whether two distinct symbolic derivations have the same sets of solutions. We have obtained a decidability result for the subterm convergent theories.

6.2.2. Security Properties

Participants: Véronique Cortier, Laurent Vigneron.

Most previous results focus on secrecy and authentication for simple protocols like the ones from Clark & Jacob library. We explore several directions to cover more complex security properties.

Non-repudiation protocols have an important role in many areas where secured transactions with proofs of participation are necessary. Formal methods are clever and without error, therefore using them for verifying such protocols is crucial. In this purpose, in collaboration with F. Klay (France Telecom R&D), we have shown how to partially represent non-repudiation as a combination of authentications, and also defined a new method , based on the handling of the knowledge of protocol participants. This last method has been implemented in the AVISPA Tool, and used for analyzing several protocols [40].
In particular, it has been used with L. Jing (Sun Yat-Sen University, China) for defining and analyzing a non-repudiation protocol for which there is no assumption of existence of resilient channels between the TTP and each protocol participant [20].
Our method has also been used with Ambuj Pushkar Ojha (INRIA Internship, from IIT Bombay, India) for modeling the a protocol defined by Cederquist, Dashti and Mauw, and analyzing it, finding fairness attacks.

.

Several security cannot be defined (or cannot be naturally defined) as trace properties and require the notion of *observational equivalence*. Typical examples are anonymity, privacy related properties or statements closer to security properties used in cryptography. In the context of the applied pi calculus and for *determinate* processes, we have shown [32] that observational equivalence actually coincides with trace equivalence, a notion simpler to reason with. Most existing protocols can actually be shown to be determinate. Then, for determinate processes without replication, we deduce decidability of observational equivalence for a general class of equational theories, reducing the decidability of trace equivalence to deciding an equivalence relation introduced by M. Baudet.

6.2.3. Advanced Classes of Protocols

Participants: Mathilde Arnaud, Najah Chridi, Véronique Cortier, Michaël Rusinowitch, Mathieu Turuani, Laurent Vigneron.

New classes of protocols are still emerging and not all can be analysed using existing techniques. We study how to cover the emergent families of security protocols.

Group Protocols. Although many works have been dedicated to standard protocols, very few address the more challenging class of group protocols. We have investigated group protocol analysis in a synchronous model, that allows the specification of unbounded sets of agents with related behavior. Also, when used in an asychronous way, this generalizes standard protocol models with bounded number of agents by permitting unbounded lists inside messages (including unbounded number of variables, nonces, etc..). This approach also applies to analyzing Web services manipulating sequences of items. In this model we propose a decision procedure for the sub-class of well-tagged protocols with autonomous keys. [10], [30].

In collaboration with the MADYNES EPI, and in the framework of SAFECAST project on secured group communication system design, we have experienced the use of UML and two complementary verification tools [45]: AVISPA enabled us detecting and fixing security flaws; the TURTLE toolkit enabled us saving development time by eliminating design solutions with inappropriate temporal parameters.

Securing routing Protocols. The goal of routing protocols is to construct valid routes between distant nodes in the network. If no security is used, it is possible for an attacker to disorganize the network by maliciously interacting with the routing protocols, yielding invalid routes to be build. That is why secure versions of routing protocols are now developed. Mathilde Arnaud has recently started a PhD, in collaboration with the project-team SECSI (LSV, Cachan) on designing verification techniques adapted for routing protocols. In particular, she has proposed [24] a new model and an associated decision procedure to check whether a routing protocol can ensure that honest nodes only accept valid routes, even if one of the nodes of the network is compromised. This result has been obtained for a bounded number of sessions, adapting constraint solving techniques.

Security APIs. In some systems, it is not possible to trust the host machine on which sensitive codes are executed. In that case, security-critical fragments of a program should be executed on some tamper resistant device (TRD), such as a smartcard, USB security token or hardware security module (HSM). The exchanges between the trusted and the untrusted infrastructures are ensured by special kind of API (Application Programming Interface), that are called *security APIs*. We have proposed [33], [56] a new and generic API that can be used to implement most key-exchange protocols on untrusted host machines.

6.2.4. Securely Composing Protocols

Participants: Stefan Ciobaca, Véronique Cortier.

Even when a protocol has been proved secure, there is absolutely no guarantee if the protocol is executed in an environment where other protocols, possibly sharing some common identities and keys like public keys or long-term symmetric keys, are executed. In [17], we show that whenever a protocol is secure, it remains secure even in an environment where arbitrary protocols are executed, provided each encryption contains some tag identifying each protocol, like e.g. the name of the protocol.

Protocols may also be built in a modular way. For example, authentication protocols may assume pre-distributed keys or may assume secure channel. How security of these protocols can be combined is an important issue. Stefan Ciobaca has started a PhD on this subject this year, in collaboration with the project-team SECSI (LSV, Cachan).

6.2.5. Soundness of the Dolev-Yao Model

Participant: Véronique Cortier.

All the previous results rely on symbolic models of protocol executions in which cryptographic primitives are abstracted by symbolic expressions. This approach enables significantly simple and often automated proofs. However, the guarantees that it offers have been quite unclear compared to cryptographic models that consider issues of complexity and probability. Cryptographic models capture a strong notion of security, guaranteed against all probabilistic polynomial-time attacks.

A recent line of research consists in identifying when it is possible to obtain the best of both cryptographic and formal worlds in the case of public encryption: fully automated proofs and strong, clear security guarantees. We have proposed a survey [55] of the results obtained so far. Moreover, we have proposed a framework and proof techniques to identify when static equivalence can be used for proving indistinguishability of bitstrings [15].

6.2.6. Safe and Efficient Strategies for Updating Firewall Policies

Participants: Abdessamad Imine, Michaël Rusinowitch.

The large size and complexity of modern networks result in large and complex firewall policies. Two policy editing languages, Type I and Type II, are generally used to update the firewall policies. Due to intervening nature of firewall rules, correct configuration and *deployment* of large policies is a difficult and error-prone task. We have shown that some recently proposed deployment algorithms in the network security contain seriousflaws [52]. Then we have defined a notion of safe deployment strategies. We have provided linear algorithms for Type I safe deployment and an approximatively linear and safe algorithm for Type II.

6.3. Model-based Verification

We have investigated extensions of regular model-checking to new classes of rewrite relations on trees. We have studied specification and proof of modular imperative programs.

6.3.1. Safety Verification Techniques with Regular Fixpoint Computations

Participants: Roméo Courbis, Pierre-Cyrille Héam, Olga Kouchnarenko.

Term rewriting systems are now commonly used as a modelling language for programs or systems. On those rewriting based models, reachability analysis, i.e. proving or disproving that a given term is reachable from a set of input terms, provides an efficient verification technique. Many recent works have shown the relevance of regular approximation techniques to tackle in practice undecidable reachability problems.

In [67], we address the following general problem of tree regular model-checking: decide whether $R^*(L) \cap L_p = \varnothing$ where R^* is the reflexive and transitive closure of a successor relation induced by a term rewriting system R, and L and L_p are both regular tree languages. We develop an automatic approximation-based technique to handle this – undecidable in general – problem in most practical cases, extending an over-approximation approach initially developed in [73] to check the reachability of terms. Moreover, we also show in [37] how the approach can be used to compute under-approximations. We also make approximation-based approach fully automatic for practical validation of security protocols. In particular, the technique was successfully used to detect attacks in security protocols like NSPK-xor, DH-exp.

To check reachability of particular terms, we improved the over-approximation approach above. Given a term t, we try to compute an over-approximation which does not contain t by refining the approximation. If the approximation refinement fails then t is a reachable term. The above technique has been extended to left-linear term rewriting systems. However it requires to perform some determinisation steps with an exponential time and space complexity, and it is therefore practically unfeasible. In [16], we address this problem for non-left linear rules by proposing an algorithm replacing determinisation (exponential steps) by polynomial time constructions on involved automata. This algorithm is a generalisation of the algorithm presented in [65] which addresses the problem of left-quadratic rules. It should be noticed that many industrial specifications give rise to non-left linear rules, like in security protocols analysis, or in backward analysis of Java bytecode.

To go further, we propose in [35], to exploit rewriting approximations for model-checking of linear time temporal properties. We show the helpfulness of the reachability analysis for model-checking three useful temporal property patterns on infinite state rewriting graphs. The reachability problem being in general undecidable on non terminating TRSs, we provide a construction based on tree automata with global equalities and disequalities (TAGED for short), and then design approximation-based semi-decision procedures to model-check useful temporal patterns on infinite state rewriting graphs. To show that the above TAGED-based construction can be effectively carried out, complexity analysis for rewriting TAGED-definable languages is given. A deep integration of our proposals in the model-checking process is achieved by using the same over-approximations for computing (finite representations of) some parts of the infinite state model, as for verifying linear time temporal properties.

6.3.2. Random Generation of Tree Automata

Participant: Pierre-Cyrille Héam.

The widespread use of automata as primitive bricks in verification motivates an ever renewed search for efficient algorithms taking automata as input. Developing new algorithms and heuristics raises crucial evaluation issues, as improved worst-case complexity upper-bounds do not always transcribe into clear practical gains. A suite for software performance evaluation can usually gather three types of entries:[5]

1. benchmarks, i.e. large sets of typical samples, which can be prohibitively difficult to collect, and thus only exist for a few general problems,
2. hard instances, that provide good estimations of the worst case behaviour, but are not always relevant for average case evaluations,
3. random inputs, that deliver average complexity estimations, for which the catch resides in obtaining a meaningful random distribution (for instance a uniform random distribution). As the mathematical computation of the average complexity of an algorithm is an intricate task that cannot be undertaken in general, random inputs can prove themselves invaluable for its empirical estimation.

We present in [38] a general rejection algorithm that uniformaly generates sequential letter-to-letter transducers up to isomorphism. We tailor this general scheme to randomly generate deterministic tree walking automata and deterministic top-down tree automata. We apply our implementation of the generator to the estimation of the average complexity of a deterministic tree walking automata to nondeterministic top-down tree automata construction we also implemented. Overall, the translation results in a $\mathcal{O}(2^n)$ size increase on average, which is significantly better than the worst-case $\mathcal{O}(2^{n2})$ bound.

6.3.3. Integer Weighted Automata Positivity Problem

Participants: Pierre-Cyrille Héam, Olga Kouchnarenko.

Weighted automata is a formalism widely used in computer science for applications in image compression, speech-to-text processing or discrete event systems. These large application areas make them intensively studied from the theoretical point of view. The expressive power of these automata is high enough so that many natural problems are not decidable. Among them the problem to know whether for a given integer weighted automaton $\mathcal{A}$, every word has a positive cost, called the positivity problem, was shown to be undecidable [74]. This problem is of special interest because systems/components comparisons modelled by integer weighted automata can be based on or reduced to it.

In [27], we translate the above problem into a reachability problem and investigate two semi-decision approaches to tackle it. The first approach is based on a configuration space exploration using a pruning property to reduce the search. The second approach uses a rewriting encoding of the problem and applies approximation techniques developed in the rewriting theoretical framework. These two approaches have been implemented and tested on random inputs providing promising results.

6.3.4. Modular Specification of Imperative Programs

Participants: Alain Giorgetti, Olga Kouchnarenko, Elena Tushkanova.

A well conceived program is developed in a modular way, that is by the structured assembly of simpler components. A challenge is to get modularity to specify and prove modular imperative programs. It is one of the objectives of the INRIA CeProMi “Action de Recherche Collaborative”(ARC). In [61] we have illustrated the subject by specifying an algorithm to sort a Java array. An INRIA research report on all the CASSIS contributions to the CeProMi project will be made public at the end of 2009. Some of the work done prepares a joint (intra Cassis, Nancy-Besançon) work on “Specification and formal certification of (combination of) decision procedures”.

6.4. Model-based Testing

Our advances in Model-Based Testing (MBT) are related to extending the MBT approach to the Web Services validation. They also involve test generation from scenarios using symbolic animation of model, and test generation using formal properties. Experience of team in MBT and practice is developed in book [49] published this year.

[5] All of the three types are used in SAT-solver competitions like http://www.satcompetition.org/.

6.4.1. *Test Generation from Behavioral Models*

Participants: Fabrice Bouquet, Thibaud Brocard, Pierre-Christophe Bué, Kalou Cabrera, Frédéric Dadeau, Stéphane Debricon, Alain Giorgetti, Adrien de Kermadec, Jonathan Lasalle, Vincent Pretre.

We have introduced an original model-based testing approach that takes a UML behavioural view of the system under test and automatically generates test cases and executable test scripts according to model coverage criteria. We also proposed a solution on the basis of this work to treatbusiness process testing [46]. In parallel, we are working on the improvement of the test generation technique, by combining constraint solving and theorem proving, in order to detect inconsistencies in the behaviors extracted from the model, and to find a relevant instantiation of the initial test data [21].

6.4.2. *Test Generation from Scenarios*

Participants: Fabrice Bouquet, Pierre-Christophe Bué, Kalou Cabrera Castillos, Frédéric Dadeau, Adrien de Kermadec.

In the context of the RNTL POSE project[6], the team has developed and experimented a language describing test scenarios. Basically, a scenario is a regular expression describing sequences of operations calls (without specifying their possible parameters) along with intermediate states that have to be reached. Each scenario is unfolded and played using a symbolic animation engine, that instantiates the sequence. This approach has been experimented on the IAS case study of Gemalto, and also applied on a model of the POSIX standard. The process has been implemented within a tool named jSynoPSys [19]. Our current investigations, starting in the TASCCC project (ANR 2009) will focus on the automated generation of test scenarios from a dedicated set of property patterns.

In addition, we have defined conformance relationships dedicated to establishing a verdict when testing the correct implementation of security policies (namely access control policies) in smart cards applications. These conformance relationships are variants of input-output conformance and are based on the inclusion of traces of the implementation w.r.t. traces computed on a security-dedicated model, involving possible mappings between the values of these two levels.

Also, we use scenario information to compute an abstraction of model. This abstraction can be use in two ways. The first is a referential coverage for test sequences. The second is to compute test sequences itself.

6.4.3. *Random Combination*

Participants: Frédéric Dadeau, Pierre-Cyrille Héam.

We are also beginning experiments on the combination of random- and model-based testing. A first attempt has been done to automatically produce LTL formula using uniform random test generation. More recently, an approach has considered the automated generation of automata in order to evaluate various FSM-based test generation algorithms. A major result is the highlighting of an error in a widely-spread implementation of the chinese postman algorithm. We also proposed a test generation technique, driven by a final number of test cases, and combining random testing and model-based testing. It consists in arbitrarily augmenting a FSM in order to reach a given number of test cases when selected FSM-based test generation algorithms are applied. A realistic experiment has illustrated the efficiency of this approach. These works are summarized in [18].

6.5. Verification for Service Oriented Computing

We have investigated several specific verification problems related to the composition of services including security issues and quality of service.

6.5.1. *Towards An Automatic Analysis of Web Services Security*

Participants: Tigran Avanesov, Yannick Chevalier, Mohamed Anis Mekki, Michaël Rusinowitch, Mathieu Turuani, Laurent Vigneron.

[6] http://www.rntl-pose.info

Automatic composition of web services is a challenging task. Many works have considered simplified automata models that abstract away from the structure of messages exchanged by the services. For the domain of secured services (using e.g. digital signing or timestamping) we propose a novel approach to automated orchestration of services under security constraints. Given a community of services and a goal service, we reduce the problem of composing the goal from services in the community to a security problem where an intruder should intercept and redirect messages from the service community and a client service till reaching a satisfying state. This work has been pursued in the context of AVANTSSAR FP7 project.

6.5.2. Composition of Web Services

Participants: Christophe Ringeissen, Laurent Vigneron.

In collaboration with Olivier Perrin (LORIA) and Eric Monfroy (UTFSM Valparaíso, Chile), we are working on applying constraint programming techniques to the composition problem. Our approach consists in instantiating a given abstract representation of a composite Web service by selecting the most appropriate concrete Web services. This instantiation is based on constraint programming techniques which allows us to match the Web services according to a given request. Our proposal performs this instantiation in a distributed manner, i.e., the solvers for each service type are solving some constraints at one level, and they are forwarding the rest of the request (modified by the local solution) to the next services. When a service cannot provision part of the composition, a distributed backtrack mechanism enables to change previous solutions.

6.5.3. Controlling Access in Distributed Collaborative Editors

Participants: Asma Berregba, Abdessamad Imine.

Distributed Collaborative Editors (DCE) belong to a particular class of distributed systems that enables several and dispersed users to form a group for editing documents (*e.g.* Google Wave). To ensure data availability, the shared documents are replicated on the site of each participating user. Each user modifies locally his copy and then sends this update to other users. Controlling access in such systems is still a challenging problem, as they need dynamic access changes and low latency access to shared documents. In this work, we propose a *flexible* access control model where the shared document and its authorization policy are replicated at the local memory of each user [57], [39]. To deal with latency and dynamic access changes, we use an optimistic access control technique in such a way that enforcement of authorizations is retroactive [29]. We show that naive coordination between updates of both copies can create security holes on the shared document, by permitting illegal modifications or rejecting legal modifications. A prototype based on our concurrency control framework has been implemented for supporting the secure and collaborative editing of HTML pages. This prototype is deployed on P2P JXTA platform.

6.5.4. Composition of services with constraints

Participants: Pierre-Cyrille Héam, Olga Kouchnarenko.

In [47], we focus on the composition of Web services with constraints. The originality of our approach consists in modeling the services by Boolean automata, i.e. finite automata extended with parametric Boolean conditions. The use of Boolean automata and of their partially syncronized products allows us to provide a theoretical study for three service composition problems – the Valuation Decision problem, the Boolean Formula Decision problem, and the Boolean Formula Synthesis problem. New complexity results are established for these problems when considering both simulation-based and trace-based relations between automata.

6.5.5. Web Services Validation

Participants: Fabrice Bouquet, Vincent Pretre.

In order to validate Web Services applications, we explore model-based testing methodologies combined with common criteria. The results of tests are used to compute a mark that qualifies the quality of web services operations. This solution is then integrated in a validation framework based on an UDDI server. In this framework, named iTac-QoS Web Services are tested when they are declared to the UDDI server, and the obtained marks are supplied to customers looking for services. We propose an original approach to take into account the composition of Web services from their models as described in [21].

7. Contracts and Grants with Industry

7.1. Research Result Transfer

The BZ-Testing-Tools technology has been transfered to LEIRIOS Technologies, at the end of 2004. The partnership between the Cassis project and the R&D LEIRIOS Department, located at the TEMIS Scientific and Industrial area at Besançon, will be continued through (national and international) projects or with a new transfer protocol. According to the law of innovation, F. Bouquet is scientific consultant of LEIRIOS Technologies.

7.2. European Projects

- AVANTSSAR — *Automated validation of trust and security of service-oriented architectures*. STREP Project funded under 7th FP (Seventh Framework Programme) Research area: ICT-2007.1.4 Secure, dependable and trusted infrastructures. The coordinator is the University of Verona (Italy) and the Cassis project is one of the 10 partners. AVANTSSAR aims to propose a rigorous technology for the formal specification and "Automated VAlidatioN of Trust and Security of Service-oriented ARchitectures". This technology will be automated into an integrated toolset, the AVANTSSAR Validation Platform, tuned on relevant industrial case studies.
- SecureChange[7] is funded under the 7th FP (Seventh Framework Programme) Research area: ICT-2007.8.6: ICT forever yours. The project will develop processes and tools that support design techniques for evolution, testing, verification, re-configuration and local analysis of evolving software. Our focus is on mobile devices and homes, which offer both great research challenges and long-term business opportunities. The project is lead by Fabio Massacci (University of Trento, Italy) and it is has started in February 2009 for a period of 36 months.

  The project is leaded by Fabio Massacci (University of Trento, Italy) and it is expected to start at the beginning of 2009 for a period of 36 months.

7.3. INTERREG

INTERREG TEST-INDUS — We are working with the university of Geneva, SMARTESTING Technologies and CLIO SA. The project concerns the test generation in industrial process. The consortium will propose methods, techniques and tools to integrate (model-based) testing into industrial process. The duration of the project is 18 months and started in May 2008.

8. Other Grants and Activities

8.1. International Grants

- Project INRIA-CNPq (Brazil), DA CAPO — *Automated deduction for the verification of specifications and programs*. This is a project on the development of proof systems (like *haRVey*) for the verification of specifications and software components. The coordinators are David Déharbe (UFRN Natal, Brazil) and Christophe Ringeissen. On the french side, DA CAPO also involves MOSEL and PAREO.
- Project INRIA-CONICYT (Chile), CoreWeb — *Constraint Reasoning for the Composition of Web Services*. The coordinators are Eric Monfroy (UTFSM Valparaíso, Chile) and Michaël Rusinowitch.

[7] http://www.securechange.eu

- Associate Team INRIA (with UTFSM Valparaíso, Chile), VanaWeb — *Hybrid and autonomous constraint solving and applications to composition problems for the Web*. The coordinators are Carlos Castro (UTFSM Valparaíso, Chile) and Christophe Ringeissen. On the french side, VanaWeb also involves members of ECOO and PAREO.
- French-Tunisian project on *Security Policies and Configurations of Firewalls: Compilation and Automated Verification*. We collaborate with SupCom Tunis and the project-team DAHU in the context of STIC-Tunisia.
- PHC Alliance project between the Cassis team and the University of Bristol on refinement of security systems. The coordinators of the projet are Bogdan Warinschi and Véronique Cortier. Duration: 2 years, started in January 2008.

8.2. National Grants

- ARA SSIA FormaCrypt—*Formal proofs and probabilistic semantics in cryptography*, duration: 3 years, started in January 2006. The verification of cryptographic protocols is a very active research area. Most works on this topic use either the computational approach, in which messages are bitstrings, or the formal approach, in which messages are terms. The computational approach is more realistic but more difficult to automate. The FormaCrypt project aims at bringing together these orthogonal approaches in order to get the best of the two worlds. Partners are: Liens (coordinator), project-team SECSI - LSV, Cachan.
- ARA SSIA ARROWS—*Safe Pointer-Based Data Structures: A Declarative Approach to their Specification and Analysis*, duration: 3 years, started in autumn 2005. The goal of this project is to develop new specification languages for programs manipulating pointers which are sufficiently precise to express many interesting properties and, at the same time, support automatic analyses. Partners are: CAPP-LEIBNIZ Grenoble (coordinator), LILaC-Irit Toulouse. The local coordinator is S. Ranise.
- ARA SETI RAVAJ [8] — *"Rewriting and Approximations for Java Applications Verification"*, duration: 39 months, started on January 2007. The goal of this project is to analyse MIdlets – Java programs designed for mobile devices like cell phones or PDA. In addition to classical proof tools of rewriting, we propose to use approximations of reachable terms. There are three academics partners: INRIA LANDE, INRIA PROTHEO and LIFC/Besançon; and an industrial: France Telecom R&D. The local coordinator is O. Kouchnarenko.
- ANR SESUR AVOTÉ—*Formal Analysis of Electronic-Voting protocols*, duration: 4 years, started in January 2008. Electronic voting promises the possibility of a convenient, efficient and secure facility for recording and tallying votes. However, the convenience of electronic elections comes with a risk of large-scale fraud. The AVOTÉ project aims at proposing techniques for formally analyzing e-voting protocols. The coordinator of the project is the Cassis team. Partners are: France Telecom Lannion, LSV Cachan, Verimag Grenoble.
- ANR program "Systèmes interactifs et robotique"— *Smart Surface*, coordinated by AS2M (Automatique et Systèmes Micro-Mécatroniques) department at the FEMTO-ST (Franche-Comté Electronique Mecanique Thermique et Optique - Sciences et Technologies) institute (UMR 6174). This project started in July 2007 for three years. The CASSIS participant is A. Giorgetti.
- ANR DECERT — *Deduction and Certification*, coordinated by Th. Jensen (IRISA). This project focuses on the design of decision procedures, in particular for fragments of arithmetic, and their integration into larger verification systems, including skeptical proof assistants. Partners are: IRISA Rennes, LRI Orsay, INRIA Sophia, Systerel and CEA. From INRIA Nancy, MOSEL and CASSIS project-teams are involved. This project will start in January 2009 for three years.

[8] http://www.irisa.fr/lande/genet/RAVAJ/index.html

- ANR TASCCC *Test Automatic basé sur des Scenarios et Critères Communs – Automated Testing based on Scenarios and Common Criteria*, duration: 3 years, starting in Dec. 2009. The project aims at completing the model-based testing process initiated in the POSE project, using scenarios to specify the test cases that have to be generated by model animation. The goal is here to provide an automated mean for generating the scenarios from a given set of properties. The overall objective is to ease the Common Criteria evaluation of secure softwares. Partners: :Gemalto (leader), LIG, LIFC, Supelec, Smartesting, and Serma Technologies.
- FCE Vetess [9] — We are working with the university of Haute Alsace, SMARTESTING Technologies and PSA Citroen. The project is labelled by "pole de compétitivité Véhicule du Futur" and funded by the "Fonds de Compétitivité des Entreprises", an inter-ministry grant. It aims at verifying embedded systems vehicles by automatic model-based tests generation. The duration of the project is 18 months and started in September 2008.
- Collaborative Research Initiative INRIA, ARC CeProMi “Certification de Programmes manipulant la Mémoire”, coordinated by Claude Marché from the project-team PROVAL. This project started in 2008 for two years. The partners are the project-teams GALLIUM (François Pottier) and PROVAL (Claude Marché), and DCS Team (Marie-Laure Potet, Verimag, Grenoble). The local coordinator is Alain Giorgetti.
- DGA RIE Secure Test project, duration: 18 months, started in February 2009. The project provides a specific environment to verify of cryptographic components (hardware or software) with an Model-Based Testing approach. The method help the test team to evaluation DGA to product a test refential. Partners are: DGA CELAR, Smartesting (coordinator), Telecom Bretagne. The local coordinator is F. Bouquet.

8.3. International Collaborations

- In the area of automated test generation from a formal model, we have an active collaboration with Dr Mark Utting from the Formal Method group from the University of Waikato [10]. This cooperation is supported by the France-New-Zealand scientific program.
- In the area of business applications, we are working on the soundness problem of coloured work-flow Petri nets with the Information System group of Professor K. van Hee from the Technical University of Eindhoven. This cooperation is supported by the NWO scientific program (The Netherlands).

8.4. Individual Involvement

F. Bouquet: Vice-head of LIFC laboratory, PC Member of International Conference in Soft- ware Testing (ICST’09), PC member of Modevva’09 (Model-Driven Engineering, Verification, And Validation) and PC member of MBTEST 2010.

V. Cortier: coordinator of the ANR SESUR AVOTÉ (started in January 2008); local coordinator of the ARA SSIA FormaCrypt (started in January 2006); French coordinator of the PHC Alliance project on refinement of security systems; Chair of FCS 2009 (Workshop on Foundations of Computer Security, affiliated with LICS 2009); PC member of CSF 2009 (22nd Computer Security Foundations Symposium), ESORICS’09 (14th European Symposium on Research in Computer Security), TACAS 2009 (15th International Conference on Tools and Algorithms for the Construction and Analysis of Systems), ASIAN’09 (13th Annual Asian Computing Science Conference), SARSSI’09 (Conférence sur la sécurité des architectures réseaux et des systèmes d’information), ARSPA-WITS’09 (Joint Workshop on Automated Reasoning for Security Protocol Analysis and Issues in the Theory of Security); member of the CS (Comité de sélection) for the 2009 INRIA - Rennes University chair, member of the CS (Comité de sélection) for the 2009 INRIA - ENS Cachan chair, member of the recruitment committee 2009 of junior researchers at INRIA Rocquencourt, member of the Evaluation Committee of the INRIA since September 2008.

[9] http://lifc.univ-fcomte.fr/vetess

[10] http://www.cs.waikato.ac.nz/Research/fm/index.html

F. Dadeau: PC member of the 7th International Conference on integrated Formal Methods (iFM'09), Dusseldorf, Germany. PC member of the 2nd International Workshop on Constraints in Software Testing, Verification and Analysis (CSTVA'2010), co-located with the International Conference on Software Testing (ICST'2010). Organizer of the MTVV'2009 days in Besançon (two-days workshop) on Model-Based Testing, funded by the MTVV group of the GDR GPL).
A. Giorgetti: Editorial committee member of *Techniques et Science Informatique (TSI).*
O. Kouchnarenko: director of the research team *VESONTIO* (former TFC) of the *Laboratoire d'informatique de Franche Comté (LIFC)*; PC member of "*International Workshop on Abstractions for Petri Nets and Other Models of Concurrency*", APNOC'09. Member of the "Comité de Sélection" at the "Ecole des Mines de Nancy" and the University of Nancy I. Member of the Committee INRIA CR1 and CR2 at the Center INRIA Nancy-Grand Est. Director of the "Licence Informatique 2008-2012" in the University of Franche-Comté.
C. Ringeissen: PC member of FroCoS'09 (Frontiers of Combining Systems) and IJCAR 2010 (the 5th International Joint Conference on Automated Reasoning); member of the CS (Comité de sélection) for the recruitment of assistant professors at the University Paris-Sud 11 (Orsay).
M. Rusinowitch: member of the IFIP Working Group 1.6 (Rewriting); PC member of CADE 2009 (22nd International Conference on Automated Deduction), CRiSIS 2009 (International Conference on Risks and Security of Internet and Systems), FTP 2009 (International Workshop on First-Order Theorem Proving), Luxembourg Day on Security and Reliability, SCSS 2009 (Tunisia - Japan Workshop on Symbolic Computation in Software Science), Colloque d'Informatique: Brésil / INRIA, Coopérations, Avancées et Défis, SARSSI'09 (Conférence sur la sécurité des architectures réseaux et des systèmes d'information). member of the CS (Comité de sélection) for the 2009 INRIA - Bordeaux University chair. Vice-président du comité des projets INRIA Grand Est depuis le 1/10/2009.
L. Vigneron: Member of the FTP steering committee; Member of the IFIP Working Group 1.6 on Rewriting; Webmaster of the site Rewriting Home Page and of the RTA conference site.
We are involved in several lectures of the "Master Informatique" of the universities of Nancy. L. Vigneron is in charge of the lectures on *Algorithmic verification* and *Security of communications*. V. Cortier is in charge of the lecture on *Theory of the security*. C. Ringeissen is in charge of the lecture on *Decision procedures and program verification*.

8.5. Visits of Foreign Researchers

Eric Monfroy (UTFSM Valparaíso, Chile) has visited LORIA as a Nancy 2 invited professor, to work on a constraint approach for composition of web services (June-July).
Ambuj Pushkar Ojha (IIT Bombay), NRIA Internship, (May-July).
Bogdan Warinschi (University of Bristol) has visited LORIA to work on combination techniques for soundness results of symbolic model (June 22-26th and November 16-20th).
Adel Bouhoula (SupCom Tunis) has visited LORIA from July 23 to July 28 to work on computer security.
Chris Lynch (Univ. of Clarkson) has visited LORIA in June to work on protocol verification.

8.6. Visits of Team Members

Véronique Cortier has visited Bogdan Warinschi (University of Bristol) to work on combination techniques for soundness results of symbolic models (September 27th - October 1st).
Olga Kouchnarenko has visited Natalia Sidorova (Eindhoven Univ. of Technologies) to work on refinement of may-/must workflow Petri nets (June 30th - July 14th), and on component adaptability and configuration (October 26th - November 6th).

9. Dissemination

9.1. Ph. D. Theses

Vincent Pretre has defended his Ph. D thesis (Université de Franche-Comté) entitled "Génération automatique de tests à partir de modèle formel pour les applications de type web services", on March 18, 2009.

Najah Chridi has defended her Ph. D. thesis (Université Henri Poincaré – Nancy 1) entitled “Contributions à la vérification automatique de protocoles de groupes”, on September 11, 2009.

9.2. Habilitation Theses

Pierre-Cyrille Héam has defended his habilitation (Université de Franche-Comté) entitled “Automates finis pour la fiabilité logicielle et l’analyse d’accessibilité”, on November 13, 2009.

Véronique Cortier has defended her habilitation (INPL) entitled “Analysis of cryptographic protocols: from symbolic to computational models”, on November 18, 2009.

Christophe Ringeissen has defended his habilitation (Université Henri Poincaré – Nancy 1) entitled “Equational Reasoning and Combination Methods: from programs to proofs”, on November 27, 2009.

9.3. Committees

Fabrice Bouquet is chair of Ph. D. thesis committee of Hala Sabbah (Université de Franche Comté).

Véronique Cortier is reviewer for the thesis of Mounira Kourjieh (Toulouse).

O. Kouchnarenko is a member of the ASTI 2009 committee to award the best Ph. D. dissertations [11] of the “*Fédération des Associations Françaises des Sciences et Technologies de l’Informations*.

M. Rusinowitch is reviewer for the habilitations of Thomas Genet (Rennes) and Hélène Collavizza (Nice), and for the PhD thesis of Sergiu Bursuc (Cachan) and Antoine Mercier (Cachan).

9.4. Seminars, Workshops, and Conferences

We were invited to give the following talks.

K. CABRERA CASTILLOS, *Scenario Based Testing for ensuring POSIX Compliance* (joint work with F. Dadeau, A. De Kermadec and R. Tissot). Invited Talk at the Workshop on Verified Software: Theory, Tools, and Experiments (VSTTE’2009), part of the FM’Week (Eindhoven).

V. CORTIER, Invited tutorial on Verification of Security Protocols at VMCAI’09 (Conference on Verification, Model Checking, and Abstract Interpretation), January 18th, 2009, Savannah, USA.

P. HÉAM, Invited Talk on Regular Approximations at Laboratoire Bordelais de Recherche en Informatique, Université de Bordeaux.

L. VIGNERON, Seminar on Verification of Infinite State Systems: Application to the Security Protocols Analysis, at ENS Lyon, December 8th, 2009.

10. Bibliography

Major publications by the team in recent years

[1] A. ARMANDO, S. RANISE, M. RUSINOWITCH. *A Rewriting Approach to Satisfiability Procedures*, in "Journal of Information and Computation — Special Issue on Rewriting Techniques and Applications (RTA’01)", vol. 183, n^{o} 2, June 2003, p. 140–164.

[2] F. BOUQUET, B. LEGEARD, F. PEUREUX. *CLPS-B: A Constraint Solver to Animate a B Specification*, in "International Journal of Software Tools for Technology Transfer, STTT", vol. 6, n^{o} 2, August 2004, p. 143–157.

[3] Y. CHEVALIER, L. VIGNERON. *Strategy for Verifying Security Protocols with Unbounded Message Size*, in "Journal of Automated Software Engineering", vol. 11, n^{o} 2, April 2004, p. 141–166.

[11] http://www.asti.asso.fr/prix_these_2009

[4] H. COMON-LUNDH, V. CORTIER. *Security properties: two agents are sufficient*, in "Science of Computer Programming", vol. 50, n^{o} 1-3, March 2004, p. 51–71, http://www.loria.fr/~cortier/Papiers/ComonCortierSCP03.ps.

[5] F. JACQUEMARD, M. RUSINOWITCH, L. VIGNERON. *Compiling and Verifying Security Protocols*, in "Logic for Programming and Automated Reasoning (LPAR'00), Reunion Island, France", A. VORONKOV, M. PARIGOT (editors), Lecture Notes in Computer Science, vol. 1955, Springer, 2000, p. 131–160.

[6] B. LEGEARD, F. PEUREUX. *B-Testing-Tools : génération de tests aux limites à partir de spécifications B*, in "TSI, Techniques et Sciences Informatiques, Hermès-Lavoisier", vol. 21, n^{o} 9, 2002, p. 1189–1218.

[7] B. LEGEARD, F. PEUREUX, M. UTTING. *Automated Boundary Testing from Z and B*, in "Formal Methods Europe (FME 2002)", L.-H. ERIKSSON, P. LINDSAY (editors), Lecture Notes in Computer Science, vol. 2391, Springer, 2002, p. 21–40.

[8] M. RUSINOWITCH, M. TURUANI. *Protocol Insecurity with Finite Number of Sessions and Composed Keys is NP-complete*, in "Theoretical Computer Science", vol. 299, April 2003, p. 451–475, http://www.loria.fr/~rusi/pub/tcsprotocol.ps.gz.

[9] C. TINELLI, C. RINGEISSEN. *Unions of Non-Disjoint Theories and Combinations of Satisfiability Procedures*, in "Theoretical Computer Science", vol. 290, n^{o} 1, 2003, p. 291–353.

Year Publications

Doctoral Dissertations and Habilitation Theses

[10] N. CHRIDI. *Contributions à la vérification automatique de protocoles de groupes*, Université Henri Poincaré - Nancy 1, 09 2009, http://tel.archives-ouvertes.fr/tel-00417290/en/, Ph. D. Thesis.

[11] V. CORTIER. *Analyse des protocoles cryptographiques: des modèles symboliques aux modèles calculatoires*, Institut National Polytechnique de Lorraine, 11 2009, Habilitation à Diriger des Recherches.

[12] P.-C. HEAM. *Automates finis pour la fiabilité logicielle et l'analyse d'accessibilité*, Université de Franche-Comté, 11 2009, http://tel.archives-ouvertes.fr/tel-00432301/en/, Habilitation à Diriger des Recherches.

[13] C. RINGEISSEN. *Raisonnement équationnel et méthodes de combinaison: de la programmation à la preuve*, Université Henri Poincaré - Nancy 1, 11 2009, Habilitation à Diriger des Recherches.

Articles in International Peer-Reviewed Journal

[14] H. ABDELNUR, T. AVANESOV, M. RUSINOWITCH, R. STATE. *Abusing SIP authentication*, in "Journal of Information Assurance and Security", vol. 4, n^{o} 4, 2009, p. 311-318, http://hal.inria.fr/inria-00405356/en/LU.

[15] M. BAUDET, V. CORTIER, S. KREMER. *Computationally Sound Implementations of Equational Theories against Passive Adversaries*, in "Information and Computation", vol. 207, n^{o} 4, April 2009, p. 496-520.

[16] Y. BOICHUT, R. COURBIS, P.-C. HEAM, O. KOUCHNARENKO. *Handling Left-Quadratic Rules when Completing Tree Automata*, in "International Journal of Foundations of Computer Science", vol. 20, n^{o} 5, 2009, p. 837-849, http://hal.inria.fr/inria-00427030/en/.

[17] V. CORTIER, S. DELAUNE. *Safely composing security protocols*, in "Formal Methods in System Design", vol. 34, n^{o} 1, 2009, p. 1–36, http://hal.inria.fr/inria-00332354/en/.

[18] F. DADEAU, P.-C. HÉAM, J. LEVREY. *On the Use of Uniform Random Generation of Automata for Testing*, in "Electronic Notes in Theoretical Computer Science", vol. 253, n^{o} 2, 2009, http://hal.inria.fr/inria-00429236/en/.

[19] F. DADEAU, R. TISSOT. *jSynoPSys - A Scenario-Based Testing Tool based on the Symbolic Animation of B Machines*, in "Electronic Notes in Theoretical Computer Science", vol. 253, n^{o} 2, 2009, http://hal.inria.fr/inria-00429234/en/.

[20] L. JING, L. VIGNERON. *Design and Verification of a Non-repudiation Protocol Based on Receiver-Side Smart Card*, in "IET Information Security", 2009, http://hal.inria.fr/inria-00426527/en/CN.

[21] V. PRETRE, A. DE KERMADEC, F. BOUQUET, C. LANG, F. DADEAU. *Automated UML models merging for web services testing*, in "International Journal of Web and Grid Services", vol. 5, n^{o} 2, 2009, http://hal.inria.fr/inria-00429242/en/.

Articles in National Peer-Reviewed Journal

[22] F. DADEAU, A. HADDAD, T. MOUTET. *Test fonctionnel de conformité vis-à-vis d'une politique de contôle d'accès*, in "Technique et Science Informatiques", vol. 28/4, 4 2009, http://hal.inria.fr/inria-00429230/en/.

International Peer-Reviewed Conference/Proceedings

[23] S. ANANTHARAMAN, C. LYNCH, P. NARENDRAN, M. RUSINOWITCH. *Unification Modulo Homomorphic Encryption*, in "Frontiers of Combining Systems, 7th International Symposium, FroCoS 2009, Trento, Italy", S. GHILARDI, R. SEBASTIANI (editors), Lecture Notes in Computer Science, vol. 5749, Springer, 2009, p. 100-116, http://hal.inria.fr/inria-00426798/en/US.

[24] M. ARNAUD, V. CORTIER, S. DELAUNE. *Modeling and Verifying Ad Hoc Routing Protocol*, in "Preliminary Proceedings of the 4th International Workshop on Security and Rewriting Techniques (SecReT'09), Port Jefferson, NY, USA", H. COMON-LUNDH, C. MEADOWS (editors), July 2009, p. 33-46, http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/ACD-secret09.pdf.

[25] P. BALBIANI, Y. CHEVALIER, M. EL-HOURI. *A Logical Framework for Reasoning about Policies with Trust Negotiations and Workflows in a Distributed Environment*, in "Proceedings of the 4th International Conference on Risks and Security of Internet and Systems, Toulouse, France", IEEE, 2009, p. 3-11, http://hal.inria.fr/inria-00432528/en/.

[26] M. BAUDET, V. CORTIER, S. DELAUNE. *YAPA: A generic tool for computing intruder knowledge*, in "20th International Conference on Rewriting Techniques and Applications (RTA'09), Brasília, Brazil", Lecture Notes in Computer Science, vol. 5595, Springer, June 2009, p. 148-163.

[27] Y. BOICHUT, P.-C. HEAM, O. KOUCHNARENKO. *How to Tackle Integer Weighted Automata Positivity*, in "3rd International Workshop on Reachability Problems, RP 2009, Palaiseau, France", O. BOURNEZ, I. POTAPOV (editors), Lecture Notes in Computer Science, vol. 5797, 2009, p. 79-92, http://hal.inria.fr/inria-00428998/en/.

[28] H. BOUCHENEB, A. IMINE. *On Model-Checking Optimistic Replication Algorithms*, in "29th IFIP WG 6.1 International Conference, FMOODS/FORTE 2009, Lisbon, Portugal", 2009, p. 73-89, http://hal.inria.fr/inria-00431335/en/CA.

[29] A. CHERIF, A. IMINE. *Undo-Based Access Control for Distributed Collaborative Editors*, in "Cooperative Design, Visualization, and Engineering, 6th International Conference, CDVE 2009, Luxembourg, Luxembourg", 2009, http://hal.inria.fr/inria-00431344/en/.

[30] N. CHRIDI, M. TURUANI, M. RUSINOWITCH. *Decidable Analysis for a Class of Cryptographic Group Protocols with Unbounded Lists*, in "Proceedings of the 22nd IEEE Computer Security Foundations Symposium (CSF'09), Port Jefferson, NY, USA", IEEE, 2009, p. 277-289, http://hal.inria.fr/inria-00426919/en/.

[31] V. CORTIER. *Verification of Security Protocols (invited tutorial)*, in "10th Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI'09), Savanah, USA", Lecture Notes in Computer Science, vol. 5403, Springer, January 2009, p. 5-13.

[32] V. CORTIER, S. DELAUNE. *A method for proving observational equivalence*, in "Proceedings of the 22nd IEEE Computer Security Foundations Symposium (CSF'09), Port Jefferson, NY, USA", IEEE Computer Society Press, July 2009, p. 266-276.

[33] V. CORTIER, G. STEEL. *A Generic Security API for Symmetric Key Management on Cryptographic Devices*, in "Proceedings of the 14th European Symposium On Research In Computer Security (ESORICS'09), St Malo, France", Lecture Notes in Coputer Science, vol. 5789, Springer, September 2009, p. 605-620.

[34] J.-F. COUCHOT, A. GIORGETTI, N. STOULS. *Graph Based Reduction of Program Verification Conditions*, in "Automated Formal Methods (AFM'09), colocated with CAV'09, Grenoble, France", H. SAÏDI, N. SHANKAR (editors), ACM Press, 2009, p. 40–47, http://hal.inria.fr/inria-00402204/en/, PFC (Plate-Forme de Confiance) - Pôle de compétitivité System@tic.

[35] R. COURBIS, P.-C. HÉAM, O. KOUCHNARENKO. *TAGED Approximations for Temporal Properties Model-Checking*, in "14th International Conference on Implementation and Application of Automata, CIAA 2009, Sydney, Australia", S. MANETH (editor), Lecture Notes in Computer Science, vol. 5642, Springer, July 2009, p. 135-144, http://hal.inria.fr/inria-00380048/en/.

[36] E. GIOAN, S. BURCKEL, E. THOMÉ. *Mapping Computation with No Memory*, in "8th International Conference on Unconventional Computation - UC09, Ponta Delgada, Portugal", Springer, 2009, 15, http://hal-lirmm.ccsd.cnrs.fr/lirmm-00395080/en/.

[37] P.-C. HEAM, O. KOUCHNARENKO, Y. BOICHUT. *Tree Automata for Detecting Attacks on Protocols with Algebraic Cryptographic Primitives*, in "Joint Proceedings of the 8th, 9th, and 10th International Workshops on Verification of Infinite-State Systems (INFINITY), Lisbon, Portugal", vol. 239, Electronic Notes in Theoretical Computer Science, 2009, http://hal.inria.fr/inria-00429356/en/.

[38] P.-C. HÉAM, C. NICAUD, S. SCHMITZ. *Random Generation of Deterministic Tree (Walking) Automata*, in "14th International Conference on Implementation and Application of Automata - CIAA 2009 Implementation and Application of Automata, Sydney, Australia", S. MANETH (editor), vol. 5642, Springer-Verlag, 2009, p. 115–124, http://hal.inria.fr/inria-00408316/en/.

[39] A. IMINE, A. CHERIF, M. RUSINOWITCH. *A Flexible Access Control Model for Distributed Collaborative Editors*, in "Secure Data Management, 6th VLDB Workshop, SDM 2009, Lyon, France", 2009, http://hal.inria.fr/inria-00431341/en/.

[40] F. KLAY, L. VIGNERON. *Automatic Methods for Analyzing Non-repudiation Protocols with an Active Intruder*, in "Formal Aspects in Security and Trust, 5th International Workshop, FAST 2008, Malaga, Spain, October 9-10, 2008, Revised Selected Papers", P. DEGANO, J. D. GUTTMAN, F. MARTINELLI (editors), Lecture Notes in Computer Science, vol. 5491, Springer, 2009, p. 192-209, http://hal.inria.fr/inria-00376450/en/.

[41] O. KOUCHNARENKO, N. SIDOROVA, N. TRCKA. *Petri Nets with May/Must Semantics*, in "Concurrency, Specification, and Programming, Kraków-Przegorzały, Poland", vol. 1, Humboldt University, 2009, p. 291-302, http://hal.inria.fr/inria-00426835/en/NL.

[42] E. NICOLINI, C. RINGEISSEN, M. RUSINOWITCH. *Combinable Extensions of Abelian Groups*, in "Proc. of 22nd International Conference on Automated Deduction, CADE-22, Montreal, Canada", R. SCHMIDT (editor), Lecture Notes in Artificial Intelligence, vol. 5663, Springer, 2009, p. 51–66.

[43] E. NICOLINI, C. RINGEISSEN, M. RUSINOWITCH. *Data Structures with Arithmetic Constraints: a Non-Disjoint Combination*, in "Frontiers of Combining Systems, 7th International Symposium, FroCoS 2009, Proceedings, Trento, Italy", S. GHILARDI, R. SEBASTIANI (editors), Lecture Notes in Artificial Intelligence, vol. 5749, Springer, 2009, p. 335–350.

[44] E. NICOLINI, C. RINGEISSEN, M. RUSINOWITCH. *Satisfiability Procedures for Combination of Theories Sharing Integer Offsets*, in "Proc. of 15th International Conference on Tools and Algorithms for the Construction and Analysis of Systems, TACAS 2009, York, UK", S. KOWALEWSKI, A. PHILIPPOU (editors), Lecture Notes in Computer Science, vol. 5505, Springer, 2009, p. 428–442.

[45] P. SAQUI-SANNES, T. VILLEMUR, B. FONTAN, S. MOTA, M. S. BOUASSIDA, N. CHRIDI, I. CHRISMENT, L. VIGNERON. *UML Modeling and Formal Verification of Secure Group Communication Protocols*, in "Second IEEE international workshop UML and Formal Methods, Rio de Janeiro, Brazil", 2009, http://hal.inria.fr/inria-00429747/en/, 6 pagesMX.

National Peer-Reviewed Conference/Proceedings

[46] S. DEBRICON, F. BOUQUET, B. LEGEARD. *From Business Processes to Integration Testing*, in "Actes des 5èmes journées sur l'Ingénierie Dirigée par les Modèles, Nancy", O. ZENDRA (editor), vol. 1, LORIA, 2009, http://hal.inria.fr/inria-00430539/en/.

Workshops without Proceedings

[47] P. BALBIANI, F. CHEIKH, P.-C. HEAM, O. KOUCHNARENKO. *Composition of services with constraints*, in "Formal Aspects of Component Software, Eindhoven, The Netherlands", 2009, http://hal.archives-ouvertes.fr/hal-00429876/en/.

Scientific Books (or Scientific Book chapters)

[48] C. ARORA, M. TURUANI. *Validating Integrity for the Ephemerizer's Protocol with CL-Atse*, in "Formal to Practical Security: Papers Issued from the 2005-2008 French-Japanese Collaboration", Lecture Notes in Computer Science, vol. 5458, Springer, 2009, p. 21–32.

[49] B. LEGEARD, F. BOUQUET, P. NATACHA. *Industrialiser le test fonctionnel*, Management des systèmes d'information, Dunod, 2009, http://hal.inria.fr/inria-00430538/en/.

Books or Proceedings Editing

[50] V. CORTIER, C. KIRCHNER, M. OKADA, H. SAKURADA (editors). *Formal to practical Security*, Lecture Notes in Computer Science, Springer, vol. 5458, Springer, 2009 JP .

Research Reports

[51] M. AHMAD, S. BURCKEL. *Sequential decomposition of operations and compilers optimization*, 2009, http://hal.inria.fr/inria-00428722/en/, Research Report.

[52] Z. AHMED, A. IMINE, M. RUSINOWITCH. *Safe and Efficient Strategies for Updating Firewall Policies*, 2009, http://hal.inria.fr/inria-00381778/en/, RR-6940, Research Report.

[53] M. BERRIMA, N. BEN RAJEB, V. CORTIER. *Deciding knowledge in security protocols under some e-voting theories*, 2009, http://hal.inria.fr/inria-00375784/en/, RR-6903, Research ReportTN.

[54] Y. CHEVALIER, M. RUSINOWITCH. *Compiling and securing cryptographic protocols*, 2009, http://hal.inria.fr/inria-00426669/en/, Research Report.

[55] V. CORTIER, S. KREMER, B. WARINSCHI. *A Survey of Symbolic Methods in Computational Analysis of Cryptographic Systems*, 2009, http://hal.inria.fr/inria-00379776/en/, RR-6912, Research ReportGB.

[56] V. CORTIER, G. STEEL. *Synthesising Secure APIs*, 2009, http://hal.inria.fr/inria-00369395/en/, RR-6882, Research Report.

[57] A. IMINE, A. CHERIF, M. RUSINOWITCH. *An Optimistic Mandatory Access Control Model for Distributed Collaborative Editors*, 2009, http://hal.inria.fr/inria-00381941/en/, RR-6939, Research Report.

[58] F. JACQUEMARD, M. RUSINOWITCH. *Rewrite based Verification of XML Updates*, 2009, http://hal.inria.fr/inria-00408162/en/, RR-7007, Research Report.

[59] E. NICOLINI, C. RINGEISSEN, M. RUSINOWITCH. *Combinable Extensions of Abelian Groups*, 2009, http://hal.inria.fr/inria-00383041/en/, RR-6920, Research Report.

[60] E. NICOLINI, C. RINGEISSEN, M. RUSINOWITCH. *Data Structures with Arithmetic Constraints: a Non-Disjoint Combination*, 2009, http://hal.inria.fr/inria-00397080/en/, RR-6963, Research Report.

Other Publications

[61] E. TUSHKANOVA, A. GIORGETTI, O. KOUCHNARENKO. *Specifying and Proving a Sorting Algorithm*, 2009, http://hal.archives-ouvertes.fr/hal-00429040/en/.

References in notes

[62] F. BAADER, K. U. SCHULZ. *Unification in the Union of Disjoint Equational Theories: Combining Decision Procedures*, in "Journal of Symbolic Computation", vol. 21, n^{o} 2, February 1996, p. 211–243.

[63] F. BELLEGARDE, C. DARLOT, J. JULLIAND, O. KOUCHNARENKO. *Reformulation: a Way to Combine Dynamic Properties and Refinement*, in "International Symposium Formal Methods Europe (FME 2001)", LNCS, vol. 2021, Springer-Verlag, 2001.

[64] E. BERNARD, B. LEGEARD, X. LUCK, F. PEUREUX. *Generation of Test Sequences from Formal Specifications: GSM 11-11 Standard Case-Study*, in "International Journal on Software Practice and Experience", vol. 34, n^{o} 10, 2004, p. 915–948.

[65] Y. BOICHUT, R. COURBIS, P.-C. HEAM, O. KOUCHNARENKO. *Handling Left-Quadratic Rules when Completing Tree Automata*, in "2nd Workshop on Reachability Problems - RP'08, Electronic Notes in Theoretical Computer Science, Liverpool, UK", V. HALAVA, I. POTAPOV (editors), Elsevier Science Publishers, 2008, http://hal.inria.fr/inria-00329900/en/.

[66] Y. BOICHUT, P.-C. HÉAM, O. KOUCHNARENKO. *Vérifier automatiquement les protocoles de sécurité*, in "Techniques de l'ingénieur", October 2007, p. RE95-1–RE95-8.

[67] Y. BOICHUT, P.-C. HEAM, O. KOUCHNARENKO. *Approximation based tree regular model checking*, in "Nordic Journal of Computing", vol. 14, 2008, p. 216-241, http://hal.inria.fr/inria-00429345/en/.

[68] F. BOUQUET, B. LEGEARD. *Reification of Executable Test Scripts in Formal Specification-Based Test Generation: The Java Card Transaction Mechanism Case Study*, in "Formal Methods, FME 2003", vol. 2805, Springer-Verlag, September 2003, p. 778–795.

[69] F. BOUQUET, B. LEGEARD, F. PEUREUX. *CLPS-B - A Constraint Solver for B*, in "International Conference on Tools and Algorithms for Construction and Analysis of Systems, TACAS2002, Grenoble, France", Lecture Notes in Computer Science, vol. 2280, Springer, April 2002, p. 188–204.

[70] Y. CHEVALIER, D. LUGIEZ, M. RUSINOWITCH. *Towards an Automatic Analysis of Web Service Security*, in "Proceedings of the 6th International Symposium on Frontiers of Combining Systems (FroCoS'07), Liverpool, UK", F. WOLTER (editor), Lecture Notes in Artificial Intelligence, vol. 4720, Springer, September 2007, p. 133-147.

[71] V. CORTIER, S. DELAUNE, P. LAFOURCADE. *A Survey of Algebraic Properties Used in Cryptographic Protocols*, in "Journal of Computer Security", vol. 14, n^{o} 1, 2006, p. 1–43, http://www.loria.fr/~cortier/Papiers/survey.ps.

[72] S. EVEN, O. GOLDREICH. *On the Security of Multi-Party Ping-Pong Protocols*, in "IEEE Symposium on Foundations of Computer Science", 1983, p. 34-39, http://citeseer.ist.psu.edu/46982.html.

[73] G. FEUILLADE, T. GENET, V. V. T. TONG. *Reachability Analysis over Term Rewriting Systems*, in "J. Autom. Reasoning", vol. 33, n^{o} 3-4, 2004, p. 341-383.

[74] D. KROB. *The Equality Problem for Rational Series with Multiplicities in the Tropical Semiring is Undecidable*, in "Internatioanl Journal of Algebra and Computation", vol. 4, n^{o} 3, 1994.

[75] G. T. LEAVENS, A. L. BAKER, C. RUBY. *JML: a Java Modeling Language*, in "Formal Underpinnings of Java Workshop (at OOPSLA '98)", October 1998.

[76] M. TURUANI. *The CL-AtSe Protocol Analyser*, in "Term Rewriting and Applications - Proc. of RTA, Seattle, WA, USA", Lecture Notes in Computer Science, vol. 4098, 2006, p. 277–286.