Sequoia

Security properties, process equivalences and automated verification

Contents

1 Summary of the project

The rise of the Internet and the ubiquity of electronic devices have changed our way of life. Many previously face-to-face and paper transactions nowadays have digital counterparts: home banking, electronic commerce, e-voting, etc. This digitalization of the world comes with tremendous risks for our security and privacy. In order to protect electronic transactions security protocols are deployed. These are concurrent programs that make use of cryptographic primitives, such as encryption or digital signatures, to ensure the security of data, e.g. guarantee the confidentiality of a credit card number or authentication credentials. However, design errors in such protocols can be exploited, and given the very nature of the services provided this may have important socio-economic consequences: legally binding political elections using Internet voting, have recently been deployed in several European countries (Estonia, France, Norway and Switzerland); in France 37,7 billion euros have been spent through e-commerce in 2011; and bank fraudsters are aiming to steal ever-higher amounts (with single transfers over 50,000 euros) using fully automated attack tools.

These examples demonstrate how essential it is to have solid foundations to carefully analyse and design modern security protocols. One extremely successful approach has been symbolic security protocol analysis, based on techniques from model-checking, automated reasoning and concurrency theory. For example, while designing a formal model of Google's Single Sign-On protocol - that allows a user to identify himself only once and then access various applications (such as Gmail or Google calendar) - Armando et al. discovered that a dishonest service provider could actually impersonate any of its users at another service provider. Many automated tools - e.g. ProVerif, AVISPA, Scyther, FDR/CASPER or MaudeNPA - exist for formally verifying authentication and confidentiality properties. However, the growing complexity of Internet services and the ubiquity of connected electronic devices change both the goals of security protocols, i.e. what security properties they need to achieve, and the means they employ for achieving them. Recent work on privacy preserving properties has shown that they are naturally captured as indistinguishability properties. Indistinguishability is a powerful notion that enables the analysis of a wide range of security properties. It expresses the fact that an adversary is unable to distinguish between two situations, and can be naturally modelled as equivalences in process calculi that support the modeling of cryptographic primitives.

However, most protocol analysis tools are restricted to analyzing reachability properties, such as authentication and weak forms of confidentiality, while most security properties (in particular privacy preserving properties, such as anonymity and untraceability) need to be expressed in terms of some process equivalence. The increasing use of observational equivalence as a modeling tool shows the need for new tools and techniques that are able to analyze such equivalence properties. The aims of this project are

- to investigate which process equivalences-among the plethora of existing ones-are appropriate for a given security property, system assumptions and attacker capabilities;
- to advance the state-of-the-art of automated verification for process equivalences, allowing for instance support for more cryptographic primitives, relevant for case studies;
- to study protocols that use low-entropy secrets, which arise for instance in modern multi-factor authentication protocols. We believe that security properties relevant to the presence of such secrets can be naturally expressed using process equivalences;
- to apply these results to case studies from electronic voting. Recent proposals avoid to trust the client platform and make use of low-entropy secrets which are out of the scope of current analysis techniques.

2 Consortium description and summary table

2.1 Partner description and relevance, complementarity

Inria Nancy - Grand'Est The CASSIS team of INRIA Nancy has extensive expertise in verification of cryptographic protocols. Their contributions include foundational results, such as the highly cited NP completeness result for protocol insecurity for a bounded number of sessions, decision procedures for a variety of cryptographic primitives, as well as the development of the Cl-AtSe verification tool that is part of the AVISPA platform. The work of the CASSIS team also includes the application of security protocol verification techniques to a number of areas, including web services, e-voting protocols, security APIs, etc. The members of CASSIS have also expertise in computational soundness results, i.e., obtaining complexity-theoretic security guarantees through symbolic verification.

Laboratoire Spécification et Vérification (LSV, ENS Cachan) The LSV Partner is a laboratory that is specialized in verification. This covers model-checking techniques of infinite or timed systems, and also verification of security properties through the SECurité des Systèmes d'Information (SECSI) project. For instance, they have proposed specifications of security properties and developed exact techniques for the decision of security in the case of a bounded number of sessions, for various cryptographic primitives. They also designed proof methods using upper approximations based on tree automata (the H1 tool). Finally, they have an expertise in security proofs in the cryptographic model, notably through collaborations with the LORIA group.

Interdisciplinary Centre for Security, Reliability and Trust (SnT, University of Luxembourg) Peter Ryan's ApSIA (Applied Security and Information Assurance)group, part of the SnT and is associated with the LACS (Laboratory of Algorithms, Cryptology and Security) has extensive expertise in the application of process algebras to modelling and analysis of secure systems. Ryan presented the first process algebra (CSP) formulation of non-interference, and later, with S A Schneider the paper that showed that all (sensible) notions of non-interference can be characterised as suitable process equivalences. Peter Ryan initiated and led the "Modelling and Analysis of Security Protocols" research project that developed the CSP and model-checking approach to the analysis of security protocols. The group has published extensively on cryptography, cryptographic protocols, mathematical models of computer security and, most recently, high assurance voting systems. Peter Ryan is the creator of Prêt à Voter, Pretty Good Democracy (with Vanessa Teague) and OpenVote (with Feng Hoa) verifiable voting schemes. Also with Feng Hao he proposed the novel Password Authenticated Key Establishment Protocol J-PAKE.

2.2 Relevant experience of the project coordinator

Steve Kremer is a Senior Researcher (*Directeur de Recherche*) at **Inria Nancy**. His area of research is the formal analysis and design of security protocols. Among his main achievements, he led an effort to identify, formally define and analyze security properties in electronic voting protocols. These works led to the first formal definitions for these properties and strongly influenced the subsequent formal analysis of privacy properties in RFID protocols and e-auction protocols. He has published over 50 research papers in leading journals (e.g. Journal of Computer Security, Information & Computation) and highly-selective conferences in computer security, formal methods and theoretical computer science (e.g. CSF, S&P, ESORICS, IJCAR, FSTTCS, ESOP, ICALP). He is currently PC co-chair of POST'14 and the ACM SAC'14 Security Track and serves on the Steering Committees of IEEE CSF, POST and ETAPS.

Steve Kremer has been involved in several ANR projects. In particular, he has been local PI of the ANR AVOTÉ project (2008-2012) and is currently local PI of the ANR ProSe project (2010-2014). These projects have been very successful: the AVOTÉ project was for instance labelled "Projet Phare" and produced 4 prototype tools, 2 book chapters, 7 publications in leading journals and 28 publications in international conferences. Steve Kremer was also the French PI of the French-Japanese project "Cryptography and logic: Computer-checked security proofs" (2008 - 2011).

Country	Organisation	Last Name	First Name	Current position	Role and contribution	Involvement (person.months)	Requested funding to the ANR (euros)	Requested funding to the FNR (euros)
FR	Inria Nancy	Kremer	Steve	DR Inria	*National Coordinator* Design of procedures for verifying equivalence properties, weak secrets and e-voting protocols.	19.2	210.7k	0
		Cortier	Véronique	DR CNRS	Verification of equivalence properties and design of e-voting protocols.	4.8		
		Rusinowitch	Michael	DR Inria	Verification of security protocols by rewriting techniques.	4.8		
		Turuani	Mathieu	CR Inria	Tool development and case studies.	9.6		
FR	LSV	Baelde	David	MdC ENS Cachan	*Local PI* Verification of equivalence properties including algebraic properties, tool development and partial order reductions.	14.4	140.7k	0
		Comon-Lundh	Hubert	PU ENS Cachan	Verification of symbolic and computational indistinguishability properties.	9.6		
		Delaune	Stéphanie	CR CNRS	Equivalence properties, composition results, algebraic properties and weak secrets.	9.6		
LU	Uni Lu	Ryan	Peter	Professor	*National Coordinator* Process equivalences, design of e-voting protocols, symbolic and computational models for weak secrets.	8	0	279,2k

CR = *Chargé de Recherche* (junior researcher) DR = *Directeur de Recherche* (senior researcher) MdC = *Maître de Conférence* (assistant professor) PU = *Professeur des Universités* (full professor)

3 Evolution with respect to the pre-proposal

There is no significant change compared to the pre-proposal. The consortium and work programme remain as described in the pre-proposal.

4 Context, positioning and objectives of the proposal

4.1 Context

Our society is evolving towards one in which more and more services are provided remotely, through Internet applications, and via devices ranging from desktop computers to smartphones. This rise of the Internet and the ubiquity of electronic devices have deeply changed our way of life. Digital services are no longer limited to simple e-mail accounts; most face-to-face and paper transactions nowadays have digital counterparts. They include extremely varied flavors of online banking, commerce, social networking, even services pertaining to the fundamental rights and duties of individuals, such as voting.

The stakes in all of these examples are extremely high. In e-commerce for instance, the finances of individuals, corporations, and institutions are at risk. Given the economic force that e-commerce is beginning to represent, large scale system attacks are conceivably no longer confined to damaging any one entity, but may affect and disrupt markets as a whole. Concerning social networking, arguably the most significant threat is that made to people's privacy and level of control over the distribution of personal data. The combination of people's ability to remain constantly connected and ill-conceived systems potentially opens a back door for the world into people's private lives. Finally, the prospect of tampering with individuals' decisions regarding their basic rights as members of a society is chilling, as it endangers that society's very bedrock. The right to vote and the trust placed by all in the voting process, particularly its outcome, form together the cornerstone of democracy. While the concept of e-voting offers the possibility of greatly extending these notions for all who believe in them, it likewise massively exposes them to those who wish to deceive, when improperly realized. Hence, the currently ongoing digitalization of the world comes with tremendous risks for our security and privacy. Therefore, in order to protect electronic transactions *security protocols* are deployed. Security protocols are concurrent programs that make use of cryptographic primitives, such as encryption or digital signatures, to ensure the security of data, e.g. guarantee the confidentiality of a credit card number or authentication credentials.

However, as illustrated by the following examples, design errors in such protocols can be exploited, and given the very nature of the services provided this may have important socio-economic consequences:

- *Electronic voting.* In the last few years several European countries (Estonia, France, Norway and Switzerland) organized *legally binding political elections* that allowed (part of the) voters to cast their votes remotely via the Internet. For example, in June 2012 French expatriates were allowed to vote via the Internet for parliament elections. An engineer demonstrated that it was possible to write a malware able to change the value of a casted vote without any way for the voter to notice[1]. In the 2011 Estonian parliament election, a similar attack was reported by computer scientist P. Pihelgas who conducted a real life experiment with consenting test subjects[2].

- *Financial transactions.* According to the FEVAD (the French federation of e-commerce, `www.fevad.com`), in France *37,7 billion* € have been spent through e-commerce in 2011 and Amazon's revenue in 2012 is estimated at more than *61 billion US dollars*[3]. As discussed in a white paper[4] by Dave Marcus (Director of Advanced Research and Threat Intelligence, McAfee) and Ryan Sherstobitoff (Threat Researcher, Guardian Analytics) bank fraud has changed dramatically.

[1] A video explaining the attack is available at `http://www.youtube.com/watch?v=AsvLxY478xc`

[2] The Supreme Court dismissed an electoral complaint regarding e-voting security. `http://www.nc.ee/?id=1235`

[3] `http://pdf.secdatabase.com/1562/0001193125-13-028520.pdf`

[4] Dissecting Operation High Roller. `http://www.mcafee.com/us/resources/reports/rp-operation-high-roller.pdf`

Fraudsters are aiming to steal ever-higher amounts from bank accounts (with single transfers *over 50,000 €*) and develop fully automated attack tools to do so. As a consequence, protocols need to implement more advanced, multi-factor authentication methods.

- *Privacy and traceability violations.* As reported in the New York Times[5], AOL released a list of more than 20 million web search queries. Even though the data was anonymized, users could be identified after little investigation revealing all their personal search queries. Similarly, anonymized data of social networks has been effectively used to identify persons by comparing data from several social networks[6]. Another kind of individual's privacy violation comes from the possibility to reveal their location. The use of RFID technology can be used to trace persons, e.g. in automatic toll-paying devices[7] or in public transportation. Even though security protocols are deployed to avoid tracing by third parties, protocol design errors enabled tracing of European e-passports[8]. Recently, a flaw was identified in the 3G mobile phone protocols that allows a third party, i.e., not only the operator, to trace telephones [AMR+12].

4.2 State of the art

The above examples demonstrate how essential it is to have solid foundations to carefully analyse and design modern security protocols. One extremely successful approach has been *symbolic security protocol analysis*, based on techniques from model-checking, automated reasoning and concurrency theory. This can for instance be illustrated by the following examples.

- While designing a formal model of Google's Single Sign-On protocol — that allows a user to identify himself only once and then access various applications (such as Gmail or Google calendar) — Armando *et al.* [ACC+08] discovered that a dishonest service provider could actually impersonate any of its users at another service provider. This flaw has since been corrected.

- As another example, Basin *et al.* [BCM12] have identified flaws in, and proposed fixes for, the ISO/IEC 9798 standard for entity authentication, using automated protocol verification tools. The standard has been revised based on their recommendations.

- A less recent, but famous example is the discovery of a flaw — the man-in-the-middle attack — in the Needham-Schroeder mutual authentication protocol by Lowe [Low95]. This flaw was found 17 years after the protocol was published in the context of an effort to use the CSP process algebra to formally analyze security protocols, an effort led by Peter Ryan.

The problem of automated verification of security protocols is nowadays well understood for authentication and (weak notions of) confidentiality properties, that is *reachability properties*, stating that a "bad state" cannot be reached. The general problem is known to be undecidable, while some restrictions allow to obtain decidability [DLM04]. In particular, Turuani and Rusinowitch [RT01] have shown the problem to be NP complete when the number of sessions is bounded. Results have also been obtained for various cryptographic primitives, including primitves with algebraic properties, cf. a survey by Cortier et al. [CDL06].

Moreover, automated tools — *e.g.* ProVerif [Bla01], AVISPA [ABB+05], FDR/CASPER [Low97], Scyther [Cre08], MaudeNPA [EMM09] or tamarin [SMCB12] — exist for formally verifying protocols. These tools are highly efficient and have been extensively used to analyze *reachability properties*. In particular, the CL-Atse [Tur06] tool, which is part of the AVISPA tool suite, is developed by members of the CASSIS team at Inria Nancy.

[5] A face is exposed for AOL searcher no. 4417749. The New York Times, August 9, 2006. `http://www.nytimes.com/2006/08/09/technology/09aol.html?pagewanted=all`

[6] Social sites dent privacy efforts. BBC, March 27 2009. `http://news.bbc.co.uk/2/hi/technology/7967648.stm`

[7] A Pass on Privacy? The New York Times, July 17, 2005. `http://www.nytimes.com/2005/07/17/magazine/17WWLN.html`

[8] Defects in e-passports allow real-time tracking. The Register, 26th January 2010. `http://www.theregister.co.uk/2010/01/26/epassport_rfid_weakness/`

Moreover, in the last years researchers have developed *composition results* which allow to analyze large systems in a modular way, allowing protocols to achieve security even if these protocols are executed in the presence of other arbitrary protocols. These results generally require protocols to be tagged and allow different kind of compositions, e.g. parallel [GT00, CD09], sequential [CC10] as well as self-composition [ADK08]. These composition results however only preserve reachability properties. There exist only very few composition results that preserve process equivalences. A composition result for guessing attacks, a very particular form of process equivalence, has been shown in [CDKR13]. In [ACD12], Arapinis et al. also show a composition result for equivalence properties. The hypotheses for this result to apply are however rather strong and limit the applicability of this result.

However, as shown above, **the growing complexity of modern Internet services and the ubiquity of connected electronic devices change both the goals of security protocols, *i.e.* what security properties they need to achieve, and the means they employ for achieving them.**

Recent work has shown that many security properties, in particular *privacy preserving properties* are naturally captured as *indistinguishability properties*. Indistinguishability is a powerful notion that enables the analysis of a wide range of security properties. It expresses the fact that an adversary is unable to distinguish between two situations, and can be naturally modelled as *equivalences in process calculi* that support the modeling of cryptographic primitives, such as the Spi calculus [AG99] or the applied pi calculus [AF01]. We give a few examples of security properties that have been modeled using such process equivalences.

- Ryan and Schneider [RS99] show that non-interference properties can be charactersied as suitable forms of process equivalence. Later, Kremer and Ryan [KR05] showed that anonymity in e-voting protocols can be naturally stated in terms of process equivalences. Building on this work, Delaune et al. [DKR09] show that strong versions of privacy, receipt-freeness and coercion resistance can also be expressed in terms of process equivalences. These properties have also been adapted to the case of e-auction protocols [DJP10].

- Similarly, Arapinis et al. [ACRR10] and Brusò et al. [BCdH10] have shown that untraceability in RFID protocols can be formalized as particular equivalence properties. Such definitions have been used to automatically find flaws in the European passport, and have been adapted to analyze untraceability in vehicular networks [DDS10] and in the 3G telephone protocol.

- The notion of equivalence has also been used to model semantic notions of secrecy in symbolic models, such as strong and real-or-random secrecy [Bla04]. This is in contrast to the verification of deduction based secrecy, easier to automate, and therefore more intensively studied in symbolic models. Deduction based secrecy however does not account for partial leakage of secrets and can only be seen as an approximation.

- More generally, security can be expressed as indistinguishable from an ideal system, which is correct by construction. In the Spi calculus [AG99] this idea motivated the use of process equivalence. In computational models the ideal system approach also appears in simulation-based (or universally composable) frameworks [Can01, BPW07], ideas which have recently been adapted to symbolic models as well [DKP09, BU13].

4.3 Limitations of current approaches and objectives.

Most current protocol verification tools are limited to (some forms of) confidentiality and authentication properties which are characterized as reachability properties. While authentication is naturally characterized as a reachability property, secrecy properties can only be approximated as such. Thus, if a certain symbolic value is deemed secret it is required that it never appears in the attacker's knowledge base. This is effective in many contexts but fails for example to capture notions of *semantic security*: the possibility of partial information about a data item leaking. Similarly, privacy properties such as anonymity and

untraceability have to be modelled in terms of indistinguishability. verifying these properties is however out of the scope of current protocol verification tools. Only a few exceptions exist:

- The ProVerif tool was first presented by Blanchet in [Bla01]. Since this first prototype it has been significantly developed and in particular support for some equivalence properties has been added [BAF08]. The support for verifying equivalences is however limited: ProVerif can only check equivalences for processes that only differ on messages, but have the same control flow. This limitation has been partially lifted in [CB13], but verification of e-voting protocols or the European Passport are still out of scope.

- The FDR tool [Ros94] is able to perform refinement checking of CSP processes. CSP in combination with FDR has been extensively used for verifying security protocols, see [RSG+00] for an overview. However, FDR is a general purpose verification tool and does not have native support for cryptographic primitives and is only able to analyse finite systems. Hence, both sessions and message lengths need to be artificially bounded. To allow more convenient encoding of security protocols the dedicated CASPER tool has been designed by Lowe as a front-end to FDR, but it does not provide support for equivalence properties. FDR is designed to check refinements, specifically trace and failures/divergence. It can be used to check (failures) equivalence by running refinement checks in both directions and it can directly check whether a process is determinstic (which is useful for secrecy properties that can be encoded in this way, [RWW94].

- Recently, 3 prototypes for checking equivalence properties in cryptographic pi calculi have been presented. Tiu and Dawson presented the SPEC tool [TD10] for checking open bisimulation. Cheval et al. [CCD11, Che14] presented the APTE tool for verifying trace equivalence and Chadha et al. [CCK12] the AKISS tool, also for verifying trace equivalence. Each of these tools however has shortcomings: SPEC and APTE only support a fixed set of cryptographic primitives (encryption, signatures and hashes), while AKISS supports a large class of cryptographic primitives that can be modelled by a (optimally reducing) convergent rewrite system. APTE is the only of the 3 tools that supports else branches and AKISS checks trace equivalence for a class of determinate processes (for other non-determinate processes, trace equivalence can be under- and over-approximated).

None of these tools offers support for algebraic properties, such as those appearing in exclusive or and Diffie-Hellman exponentiation.

As discussed these tools have still severe limitations. The increasing use of observational equivalence as a modeling tool shows the need for new tools and techniques that are able to analyze such equivalence properties. The aim of this project is to significantly progress the theoretical understanding and practical verification of process equivalences in the context of cryptographic protocols. We will particularly investigate the use of process equivalences in the context of protocols that use low-entropy secrets, such as out-of-band protocols, as well as in electronic voting.

More precisely the aims of this project are as follows.

- We will investigate which process equivalences — among the plethora of existing ones — are appropriate for a given security property, system assumptions and attacker capabilities. We can vary the notion of equivalence, e.g. the various flavours of bi-simulation (weak, barbed, power, loose etc), failures, testing etc to match the model of the system and its interface. We can constrain the environement to model our assumptions about the limitations of the attacker, and we can adjust the equivalence relation on the sensitive ("high") behaviours to capture various information flow policies, [Rya01]. We will also develop composition results for process equivalences that allow a modular analysis of protocols and investigate partial order reductions for such equivalences.

- We will advance the state-of-the-art of automated verification for process equivalences. This will include support for more cryptographic primitives, such as exclusive or relevant in many RFID protocols that have to ensure unlinkability to avoid people to be traced. We will also investigate how equivalence properties can be verified in computational models, as well as semi-automated

methods that allow verification when considering an unbounded number of protocol sessions. The results will be implemented in protocol verification tools and validated on case studies.

- We will study protocols that use low-entropy secrets. Such weak secrets arise for instance in modern multi-factor authentication protocols, e.g., a short, human copiable code sent by sms on a mobile phone. We believe that security properties relevant to the presence of such secrets can be naturally expressed using process equivalences. We will extend the attacker model to take into account exhaustive attacks on such weak secrets, provide tool support and validate our results on case studies.

- We will apply our results in particular to case studies from remote electronic voting. One major problem in remote e-voting is that the platforms used for voting may not be trustworthy, e.g., may be infected by a malware that leaks or even modifies a voter's choice. Recent proposals try to avoid to place trust in the client platform by using code sheets or external devices. Such protocols require voters to copy codes and therefore make use of low-entropy secrets. We will use results and tools developed in this project and formally analyze such protocols.

4.4 Relevance of the proposal and positioning

The SEQUOIA project directly contributes to the **Information and Communication Society** challenge (*3.7 Société de l'Information et de la Communication*) and in particular to the **Security of a Digital Society** part (*3.7.2.3 Axe : Sécurité de la Société Numérique*). The project is also linked to the challenge **Freedom and Security of Europe, its Citizens, and its Residents** (*3.9 Liberté et Sécurité de l'Europe, de ses Citoyens et de ses Résidents*) which includes a part on **Cybersecurity** (*3.9.5 Axe : Cybersécurité*).

At the national level there are two ANR projects in which some members of the team are involved.

- The ANR Young Researcher project VIP (ANR JCJC 2011) is led by Stéphanie Delaune and focuses on verification of indistinguishability properties. The VIP project obtained interesting preliminary results on the verification of equivalence properties. The project does however not investigate any computational security definitions, nor does it consider protocols that use weak secrets. Moreover, VIP finishes in 2015.

- The ANR VERSO ProSe (ANR VERSO 2010) is led by Bruno Blanchet (Inria Paris) and both LSV and Inria Nancy are involved. The aim of the project is to verify security protocols at the symbolic, computational and implementation level. The main focus of this project is on computational soundness and secure implementations. The project ends in 2014.

One may also mention the ERC project ProSecure, led by Véronique Cortier, member of the Cassis team. The goal of the project is to propose foundations for analysis and design of security protocols. The project will end in January 2016.

In France two other teams are working on formal methods and security protocols. The Inria Prosecco team, led by Karthik Bhargavan, is using type systems for the verification of security protocols and their implementation. This work is also sponsored by the ERC project Crysp and done in collaboration with people from Microsoft Research (cf below). In the Prosecco team B. Blanchet is also developing the ProVerif verification tool. One of the aims of this project is to overcome limitations of the ProVerif prover. The team of Yassine Lakhnech at Verimag has recently been working on formal proofs of cryptographic primitives, rather than protocols (which are more high level and make use of primitives as building blocks). These approaches are thus complementary.

At the international level several teams and projects in the area are relevant.

- The group by David Basin (ETH Zurich) is developing tools for automated verification of security protocols, e.g. the Scyther and tamarin tool. These tools are however limited to verifying reachability properties and are unable to verify equivalence properties.

- The group led by Gilles Barthe (IMDEA, Spain) is developing the EasyCrypt framework, aiming to certify game-based cryptographic proofs using the theorem prover Coq. Their approach requires much more interaction than the tools we aim to develop in this project and seeks guarantees in a computational model.
- Cédric Fournet et al. work on the F^* project at Microsoft Research. Their project develops several approaches for the verification of protocol implementations, by extraction of protocol specifications or by typing, in the symbolic and in the computational models. While their approach aims to verify implementations it requires annotations of the code.

5 Scientific and technical program, organization of the project

5.1 Scientific program and structure of the project

The project's main aims are to develop techniques for automated verification of various equivalence properties and apply these techniques to analyze protocols relying on weak secrets in out-of-band and e-voting protocols. We split the work foreseen into the following tasks.

1. Matching process equivalence to context
   - 1.1. Taxonomy of equivalences and threat contexts
   - 1.2. Scalability and Composition
2. Automated verification of process equivalences
   - 2.1. Widening the scope
   - 2.2. Beyond full automation
   - 2.3. Case studies
3. Low-entropy secrets
   - 3.1. Symbolic definitions of low-entropy secrets
   - 3.2. Computational definitions of low-entropy secrets
   - 3.3. Verification of protocols relying on low-entropy secrets
   - 3.4. Case studies
4. Application to e-voting protocols
   - 4.1. Characterising e-voting security properties
   - 4.2. Analysis of selected e-voting protocols

A detailed description of each of the tasks will be given below.

5.2 Project management

The project management will be refered to as Task 0 below.

We are planning to organize a plenary meeting every year in addition to a kick-off and final metting. Meetings will be organized, in turn, by each of the partners. We will also sometimes invite experts, external to the project, to these meetings. This will increase the visibility of the results achieved in the project at an international level and foster collaborations with other teams. Such expert may for instance include Prof. Mark Ryan (University of Birmingham, UK), Prof. Olivier Pereira (University of Louvain-la-Neuve, Belgium) and Ralf Küsters (University of Trier, Germany). Mark Ryan is an expert in formal methods and security and has already a long history of collaborations with the partners of the project. Olivier Pereira is an expert on voting systems, being one of the designers of the Helios voting system. Ralf Küsters has expertise in security definitions for electronic voting and protocol composition.

We plan the following schedule for the meetings:

December 2014	Kick-off meeting	Nancy
Fall 2015	Project meeting	Luxembourg
Fall 2016	Project meeting	Cachan
Spring 2017	Project meeting	Nancy
Spring 2018	Project meeting	Luxembourg
December 2018	Final meeting	Cachan

5.3 Detailed description of tasks

We now detail each of these research axes, organized around four work packages.

5.3.1 Task 0: Project coordination

Task 0	**Project coordination**	
Start T0 End T0+48	Duration: 48 months	Coordinator: Steve Kremer (Inria Nancy)
Objectives	Task 0 guarantees that the project evolves as planned	
Description	• Define and schedule tasks within the project • Organization of project meetings • Management of the project web page • Supervision of the progress of the work • Production of progress reports	

Deliverables	**Title**	**Date**	**Type**
D0.1	Project website	T0+1	Website
D0.2	Progress report after the first year	T0+12	Document
D0.3	Progress report after the second year	T0+24	Document
D0.4	Progress report after the third year	T0+36	Document
D0.5	Final report on project results	T0+48	Document
Validation criteria	Availability of deliverables.		

This task is in charge of guaranteeing a smooth organization of the project. It includes the organization of regular project meetings (every 6 months, as described in Section 5.2) and the delivery of progress reports.

5.3.2 Task 1: Matching process equivalence to context

Task 1	**Matching process equivalence to context**	
Start T0 End T0+36	Duration: 36 months	Coordinator: Peter Y A Ryan (Uni. Lux.)
Objectives	Compare different process equivalence and assess their usefulness for different security properties. Study compositionality of these equivalences.	
Description	See detailed description below.	

Deliverables	**Title**	**Date**	**Type**
D1.1	Taxonomy of process equivalences	T0+18	Document
D1.2	Composition issues and partial order reductions	T0+36	Document
Validation criteria	Availability of deliverables and publications.		

A key observation behind this proposal is that secrecy properties are most precisely characterized as equivalence (indistinguishability) properties. More generally, the various forms of confidentiality

properties, secrecy, anonymity, unlinkability etc., can be characterized as symmetries of certain views of the system under appropriate transformations. For example, data secrecy is captured by requiring equivalence under changes in data values and anonymity properties are captured in terms of symmetry under permutation of identities in a given pool. Richer information flow policies can be captured by introducing appropriate equivalence relations over the sensitive data or behaviours and requiring indistinguishability under symmetry transformations preserving this equivalence, [Rya01]. However, it should be noted, following [RS99], that there are many incompatible definitions of process equivalence and it is often unclear which is appropriate to any given context. The different definitions typically correspond to different models of computation and attacker capabilities. A deterministic system can be faithfully modelled using just trace equivalence, but for non-deterministic systems the situation is more subtle. For example, if the attacker is able to repeatedly run experiments on the system, each time restoring it to the same initial state, then *observational equivalence* is appropriate. Such a scenario might crop up in modelling an attacker having access to a smart card and being able to trigger a reset. On the other hand, an attacker with access to an encryption oracle is more accurately modelled by *failures equivalence*.

In short, it is essential to identify the appropriate equivalence and in this work package we will establish which notions of equivalence are appropriate to which threat context. In particular, we will identify a representative set of case studies and for each establish which forms of equivalence and environmental constraints accurately capture the notion of secrecy. The case studies should thus cover protocols employing various dedicated devices, restricted functionality, tamper evident etc., as well as the exchange of weak secrets over out-of-band channels, etc. The choice of equivalence will thus reflect the assumptions about the degree of control the adversary has in experimenting on the system, e.g. interrupting, resetting, injecting errors, accessing parts of state, etc.

Task 1.1 Taxonomy of equivalences and threats In this subtask we will identify a representative set of protocols from the domains of e-payments and e-voting and determine which equivalences notions match the different threat contexts, i.e. system assumptions about the system and about the adversary capabilities. Thus we will strive to cover a large class of categories, e.g. using TPMs, dedicated tokens, out-of-band channels etc. For these different categories we will establish which equivalence is appropriate.

This task is related to subtask 4.1, in which we survey existing e-voting protocols that employ such mechanisms to counter the client device problem. Additional protocols will emerge from a study of existing protocols for e-payments etc.

The goal of this subtask is to establish which form of process equivalence applies to which type of protocol or mechanism and threat context. Once this is done, we develop or adapt the tools accordingly to handle the required forms of equivalence and validate the approach by applying these to the representative protocols.

We note also that in some situations it will be appropriate to use an approximate form of process equivalence, either to capture some statistical or computational aspects of the system in question, or simply to render the modelling tractable for tools.

Task 1.2 Scalability and composition Most equivalence checkers for security protocols have limited applicability because they do not scale up well to analyze large systems involving several sessions running in parallel. This is unsurprising, as those systems (APTE [CCD11], SPEC [TD10], AKISS [CCK12]) are first generation tools that result from a research effort focused mostly on constraint resolution procedures and not on the concurrent nature of security protocols. The latter issue is irrelevant as far as decidability is concerned, because such tools work only for bounded numbers of sessions, but becomes criticial to obtain efficient tools. Treating concurrency in a too naive way results in a typical state-space explosion problem which makes the algorithms too ineffective as soon as several parallel processes are considered. For example, checking anonymity of Abadi and Fournet's private authentication protocol [AF04] takes only a few seconds in APTE for a single session, but this becomes minutes for two sessions and days for three. The main problem here is to treat protocols in such a way that only essential interleavings of parallel executions are considered. This classic kind of

problem has been addressed in proof search with focusing [And92] and in model-checking with partial order reduction techniques [Pel98]. In the case of reachability property, some partial-order reduction techniques have been proposed [MVB10] that are compatible with the symbolic analysis of security protocols relying on constraint resolution. Adapting those techniques to equivalence properties is a challenge, intuitively because one has to make sure that the same interleavings are ignored on the two protocols being checked for equivalence, and also because the more complex constraint resolution procedures have to be adapted to handle new kinds of constraints. We have recently obtained preliminary results for the case of simple protocols [BDH14], but a lot of work remains to be done in order to extend the scope of our technique (study larger classes of protocols, ultimately supporting non-determinism) and its practical usefulness (extend constraint resolution procedures to integrate our optimizations into existing tools).

Instead of improving the efficiency of checking protocols, it is sometimes possible to analyze large systems in a modular way. This line of research has been quite successful for analyzing reachability properties (*e.g.* [CD09, CC10]). Regarding privacy-type properties, very few composition results exist. In [ACD12], the authors identify sufficient conditions under which protocols can "safely" be executed in parallel as long as they have been proved secure in isolation. This composition result is quite general from the point of view of the cryptographic primitives but only deal with the particular case of parallel composition whereas we may want to compose protocols in different ways (*e.g.* sequential composition). Moreover, it only allows protocols that share some standard primitives provided that they are tagged. It would be useful to relax this condition in order to compose real-world protocols (and not their tagged versions) or to compose protocols that both rely on primitives for which no tagging scheme actually exists (*e.g.* exclusive-or) or, last but not least, to compose several sessions of the same protocol. Extending the scope of this result would allow one to obtain reduction results, and to obtain guarantee (vote-privacy, untraceability) in a setting that involves an arbitrary number of voters or tags studying only scenarios that involve few voters or tags.

Who does what?

Task	Participants
1.1	Lux
1.2	LSV, Inria Nancy

5.3.3 Task 2: Automated verification of process equivalences

Task 2		**Automated verification of process equivalences**		
Start	T0	Duration: 48 months	Coordinator: David Baelde (ENS Cachan)	
End	T0+48			
Objectives		Design efficient tools for automated verification of equivalence properties.		
Description		See detailed description below.		
Deliverables		**Title**	**Date**	**Type**
D2.1		Procedures supporting new features (2.1)	T0+12	Document
D2.2		Improvements of existing tools (2.1, 2.2)	T0+18	Software
D2.3		Techniques for verification of an unbounded number of sessions	T0+36	Document
D2.4		Prototype and case studies involving unbounded sessions	T0+48	Software
Validation criteria		Availability of deliverables and publications.		

Even though the notion of process equivalence has been used for specifying a wide range of security properties [RS99], most verification tools for security protocols do not support equivalence properties. A notable exception is ProVerif [Bla01], but its capabilities are restricted and it fails to conclude on many interesting examples. A few other prototype tools also exist, but each of them has restrictions

and shortcomings. The aim of this work package is to investigate theoretical aspects, such as decidability and complexity questions, as well as practical results allowing to verify protocols which are out of the scope of existing tools.

Task 2.1 Widening the scope of existing procedures We recently proposed a procedure for verifying equivalence properties [CCK12], implemented in the AKISS prototype, which will serve as a starting point. We foresee to generalize the procedure to support more cryptographic primitives, i.e. more equational theories, in particular theories reflecting algebraic properties such as those for *Diffie-Hellman exponentiation* and *exclusive or*. Currently, to the best of our knowledge no tool is able to verify equivalence properties for these theories. For this we plan to adapt techniques similar to those used in the Maude NPA [EMM09] and Tamarin [SMCB12] tools for reachability properties. However, the problem for equivalence properties is more complex. In particular ensuring termination of AKISS (which will have to perform Horn clause resolution modulo AC) will be challenging and will require the design of new resolution strategies.

We will also work to remove one limitation on the class of protocols supported by AKISS: currently the procedure used in that tool is not able to treat *protocols with else branches*. Being able to consider else branches is however crucial in many protocols that wish to ensure privacy. Typically, when invalid messages are received the protocol (enters an else branch and) needs to react in a way that does not leak side information which break privacy. For instance, in the private authentication protocol by Abadi and Fournet [AF04] a decoy message is sent in response to a message with an invalid identity to avoid repeated testing of identities. Similarly, in some versions of the European e-Passport, Chothia et al. [CS10] showed that it was possible to trace a passport due to different error messages. The APTE tool [CCD11, Che14] is able to handle else branches but the set of cryptographic primitives supported by the tool is limited. The ProVerif tool allows else branches as well, but reports false attacks on many examples (not necessarily related to the use of else branches). Indeed, the Horn clause based encoding of else branches in ProVerif seems like an interesting starting point for adding else branches to AKISS.

Finally, we will explore ways to verify *equivalence properties in the computational model*. An interesting approach here is that of Comon and Bana [BCL12] which allows to prove security properties in the computational model while using logical methods that are amenable to automation [CLCS13]. The basic idea is to prove in (fragments of) first-order logic that the security properties follows from axioms that specify what *cannot* be achieved by the (probabilistic, polynomial time) attacker — these axioms typically correspond to well-known game-based properties of cryptographic primitives in the computational model, such as IND-CPA, IND-CCA, etc. The Comon-Bana approach is relatively recent and has so far only been considered for reachability properties. The next step, that we want to explore, is to lift the theoretical framework and practical tools from reachability to equivalence-based properties.

Task 2.2 Beyond full automation Designing exact procedures for verifying equivalence properties has proven to be a difficult problem, which is known to be infeasible in general due to the complexity of protocols (replication) or primitives (equational theory). While it is interesting to investigate the decidability frontier of such problems (as is done, *e.g.* in [CCD13]) it seems that such theoretical work would mostly lead to a set of procedures very different from one another, each being applicable to a specific kind of protocol. Another approach is to forget about decidability where it is too constraining, and design semi-automated techniques relying on annotations and hints from the user. This way one can obtain techniques (and tools) that can be very efficient and apply uniformly to a wide array of protocols, including cases not known to fit in any decidable class. This is witnessed, for instance, by the Easycrypt tool [BGHB11] for certifying primitives and protocols in the computational model. Another popular semi-automated technique is the use of type systems. They have been used extensively to analyze information flow, but also more recently to verify equivalence properties [BFG+14].

We will explore various semi-automated techniques, building on existing verification techniques. We are currently studying type systems for proving equivalence properties, and we expect to be able to handle some equational theories that are currently out of the scope of fully automated tools by using

refinement types for which the verification of formulas can be outsourced to SMT solvers. Another line of work that we would like to develop is to use interactive proof techniques for establishing protocol equivalences with an unbounded number of sessions. We plan to require user input for key decisions in the equivalence-checking process but to keep using automated procedures for checking elementary steps, including checking for frame equivalence. This will require some carefully crafted language for specifying the shape of frames, expressive enough to capture invariants and simple enough to allow for automated analysis.

2.3 Case studies The aim of the above tasks is to enable the formal verification of security-protocols used in real-life application, and it is thus natural that we evaluate our algorithms and tools on concrete case studies. We will consider the e-passport application used in European countries [For04], and/or the phone application [3GP10a, 3GP10b, 3GP11]. Both rely on several sub-protocols whose specifications are available on-line. They are particularly suitable for evaluating the composition results or the semi-automated techniques that will be developed during the project (Tasks 2.2 and 2.3). We will also consider RFID protocols, such as those mentioned in [vDR08], which are used more and more in real life for payment, tracking, etc. Those protocols make extensive use of the exclusive-or operator, and will thus be natural candidates to validate our work on that primitive (Task 2.1). Of course, we also plan to apply our results on existing e-voting protocols: we have devoted Task 4 to this particular application, due to its complexity and specificities.

Who does what?

Task	Participants
2.1	LSV, Inria Nancy
2.2	LSV, Inria Nancy
2.3	LSV, Inria Nancy
2.4	LSV, Inria Nancy

5.3.4 Task 3: Low-entropy secrets

Task 3		**Low-entropy secrets**	
Start	T0	Duration: 48 months	Coordinator: Steve Kremer (Inria Nancy)
End	T0+48		
Objectives		Design new security definitions, attacker models and analysis tools for protocols based on low entropy secrets.	
Description		See detailed description below.	
Deliverables		**Title**	**Date**	**Type**
D3.1		Security definitions and attacker models for weak secrets	T0+24	Document
D3.2		Verification techniques for protocols based on weak secrets	T0+36	Document
D3.3		Tool prototype and case studies	T0+48	Software
Validation criteria		Availability of deliverables and publications, tool prototype and case studies.		

Weak secrets are secrets that are sufficiently low in entropy to be efficiently and easily used by ordinary humans for authentication purposes in the digital world. The most typical example would be a human-memorable password [Poi12]. Indeed, passwords are short and rarely very random. More recently, another type of weak secret has made an appearance: short, human copiable and/or memorable strings, e.g. transmitted to a user by SMS or generated by a personal hardware token [NR11].

The inherent weakness of these data items is what makes them user-friendly; however, this also makes it possible to enumerate them in a reasonable amount of time. Hence, it is essential that protocols employing weak secrets make it infeasible for adversaries to verify guesses. This is known as dictionary attack resistance.

There are mainly two sources of weak secrets. Either one generates short strings that are directly used in protocols or they are the result of a weak cryptographic function, e.g., one computes a hash from a strong secret and only keeps a few bits of the output to obtain a short secret that is human copiable. While in the first case an attacker may be able to directly brute-force the secret, in the second case an attacker may be able to find collisions, i.e., find a different strong secret that generates the same short output.

In either case, we believe that the most natural way to capture resistance against such attacks is through indistinguishability. The purpose of this work package is first to examine existing definitions and models and devise new ones for this purpose. We wish to do this in both the symbolic and computational frameworks in parallel. This will allow us to have consistent definitions in both approaches so as to be able to relate them. We will then develop and test automated verification techniques based on these definitions.

Task 3.1 Symbolic definitions of low-entropy secrets Some work has been done to study weak secrets in the symbolic model, starting with Lowe's paper [Low03] in which the notion of verifying a guess is formalized. An interesting recent attempt to symbolically specify and analyze protocols relying on low-entropy secrets has been carried out in the CSP framework [RSN11]. However, the modeling is restricted to a particular set of cryptographic primitives and the length of messages an adversary can send is artificially bounded (which could lead to missing attacks) to enable effective model-checking using the FDR tool. The security definition in itself is also complicated to understand and tailored to the particular verification method and protocol.

We believe that resistance to dictionary attacks can be naturally expressed as an equivalence property (as it has been done for the special case of password-based protocols [Bau05]). This has several advantages. It will lead to more general and natural definitions that are consistent with those found in the computational framework of provable security, where dictionary attack resistance is often obtained via semantic security (see the next paragraph). Indeed, hiding all partial information of a weak data item is crucial to keep it from being verified against an adversary's guess. This illustrates particularly well the need to address the limitation that consists of only *approximating* secrecy properties.

In this task, we will study symbolic definitions in parallel with task 3.2 below.

Task 3.2 Computational definitions of low-entropy secrets To date, in the provable security framework, by far the most studied protocols employing weak secrets have been those for password-authenticated key exchange (see e.g. [BM92, KOY01, CHK$^+$05, KV13], or see [Poi12] for a more extensive bibliography). These are designed in such that protocol runs using different passwords are computationally indistinguishable, which is equivalent to semantic security.

However, in the last decade protocols (not necessarily for key exchange) using short authentication strings over empirical out-of-band channels and computational models wherein to assess their security have been considered as well [NR11], mostly drawing inspiration from key exchange research. For instance, Vaudenay's security model [Vau05] is an adaptation of the popular Bellare-Rogaway[BR94] model for key exchange. Another example is Hoepman's model [Hoe05] which is based on the password-based key exchange model from [BPR00]. It has been demonstrated, e.g. see [Cre11], that the relations between key exchange security models are difficult to assess. Thus, the situation is likely the same for these out-of-band protocol security models. The situation is further complicated by the fact that the assumptions made on out-of-band channels differ across papers, e.g. Balfanz et al. [BSSW02] require only an authentic channel whereas Gehrmann et al. [GN04] require an authentic and secret one.

In this task we will survey, examine, and refine these definitions. This will be carried out in parallel with task 3.1.

Task 3.3. Verification of protocols relying on low-entropy secrets We will develop verification techniques for automated analysis of protocols that rely on weak secrets. This will require to consider an extended attacker model, based on the definitions proposed in Task 3.1. We will be integrating these methods in tools, such as Akiss [CCK12] or ProVerif [Bla01, BAF08]. These tools seem

like natural candidates as they already provide some support for verification of equivalence properties. Moreover, collisions of weak cryptographic primitives could be modelled by adding equalities to the equational theory. This would be reminiscent of the approach used in [NR11] for modeling everlasting privacy which was implemented in both these tools.

Task 3.4. Case studies We will evaluate our definitions and tools on case studies. In particular we will apply definitions and tools to the protocols discussed in the survey by Nguyen *et al.* This task will interact with tasks 3.1, 3.2 and 3.3 as it will be essential to validate our definitions and the effectiveness of the automated tools.

Who does what?

Task	Participants
3.1	INRIA Nancy, Uni Lux.
3.2	INRIA Nancy, Uni Lux.
3.3	INRIA Nancy, LSV
3.4	INRIA Nancy, LSV, Uni Lux.

5.3.5 Task 4: Application to e-voting protocols

Task 4		**Application to e-voting protocols**	
Start	T0+12	Duration: 36 months	Coordinator: Peter Y A Ryan (Univ. Lux.)
End	T+48		
Objectives		Apply the tools and techniques to a representative set of e-voting protocols.	
Description		See detailed description below.	
Deliverables		**Title**	**Date** / **Type**
D4.1		Security definitions and attacker models for e-voting protocols	T0+18 / Document
D4.2		Analyses of selected e-voting protocols	T0+36 / Document
D4.3		Publications based on the case studies	T0+48 / Document
Validation criteria		Availability of deliverables and publications.	

Electronic voting protocols are extremely complex and most attempts for formal verification to date rely on hand proofs. This is partially due to the cryptographic primitives, such as homomorphic encryption, which is out of the scope of existing automated verification tools. Also, the properties required of most voting systems are quite novel, including verifiability, receipt-freeness, coercion resistance etc. The later two are special cases of anonymity properties, hence amenable to modeling in terms of process equivalence [DKR09], but they correspond to very special attacker models. For coercion resistance for example, we assume that the attacker is free to interact with the voter at almost all steps of the protocol. Thus, the coercer can issue instructions before and during the vote casting steps and he can seek to verify that his instructions have been complied with as the protocol unfolds. Thus, for example, we can have very subtle forms of attacks such as *randomisation attacks* in which the coercer requires the encrypted receipt to have certain characteristics. Formalising such subtle properties has not yet reached consensus even at the abstract level, and has scarcely been attempted at the more concrete level of specific protocols involving dedicated devices and out-of-band channels etc.

It is worth mentioning also the notion of *software independence* proposed by Rivest and Wack, [RW06]. This is a useful notion in the evidence based approach to designing and evaluating voting systems and has been incorporated in the US technical guidelines for election technologies. It is clearly well suited to characterisation as an equivalence style property but to date it has not been formalised.

Moreover, one of the main problems in Internet voting protocols is that the client software used for voting cannot be trusted: malware may divulge the vote or other supposedly secret values such as

passwords, encryption randomisations, etc. or even simply alter it. A first solution to this problem was proposed by Chaum [Cha01] who introduced the notion of *code voting*. Instead of entering the vote itself, the voter inputs a code, representing the vote but which cannot be linked to the actual vote by the client platform. The scheme however does not offer verifiability. An improvement was proposed by Ryan and Teague in the Pretty Good Democracy [RT09] scheme. This scheme does offer verifiability, but the integrity of the election could be compromised in the case where code sheets are leaked. Another promising direction to circumvent this problem is to rely on an external hardware device with a proper display which may be used to perform some computation. Such *hardware tokens* are already used by some banks to enhance the security of online transactions. Recently there have been some proposals using such tokens in the context of e-voting, e.g. [HK14], [GRCC]. These tokens can be easily audited and, being offline, are less vulnerable to malware. In order to be effective, the token must not be connected to the computer and requires short strings to be copied by the human from the token to the computer or vice-versa. Therefore the kinds of attacks and analyses discussed in Task 3 are relevant here as well.

Another approach to countering the client device problem is to use independent channels, on the assumption that is will be harder for an adversary to compromise multiple channels and devices. Thus for example, the Norwegian e-voting system sends back confirmation codes to the voters mobile phone to confirm the correct vote reception. The independent channels can also be used to help the voter in the creation of a one-time pad encryption of her vote or to confirm the vote submission, cf. [BGS13]. Recently, Koenig et al. [KLH13] argued that the independent channels assumption may be hard to hold on practice and described an attack to the Norwegian Internet voting system.

The MarkPledge technique [Nef04], proposed by Neff in 2004 for controlled voting environments, is another approach that, using low entropy secrets, allows the vote client machine to prove to the voter that it correctly encrypted the voter's vote. The MarkPledge technique was adapted to Internet voting by Joaquim et al. [JRF09, JFR13], together with code voting and hardware tokens to enhance the privacy properties of the vote protocols. Recently, Joaquim [Joa14], has published a new cryptographic protocol based on the MarkPledge approach that is able to produce a single low entropy verification code for complex ballots. The MarkPledge voter vote verification approach is interesting because it does not requires trusted devices for end-to-end verifiability, it only requires them for the privacy properties. However, the use of the MarkPledge approach has some subtle steps and no complete formal security proofs have been made.

One of the objectives of this work package is to construct faithful models of several representative e-voting schemes along with their security properties and complete formal security proofs. In particular, we intend to construct a formal proof of a protocol satisfying both vote privacy and end-to-end verifiability without relying on trusted client software. The work will be structured into the following two subtasks.

Task 4.1. Characterising e-voting security properties In this subtask we will identify a representative set of e-voting protocols to cover the key mechanisms employed to counter the client device problem: dedicated devices, out-of-band channels with weak secrets etc. For each of these we will construct appropriate models and definitions to faithfully capture the assumptions about the system and adversary capabilities.

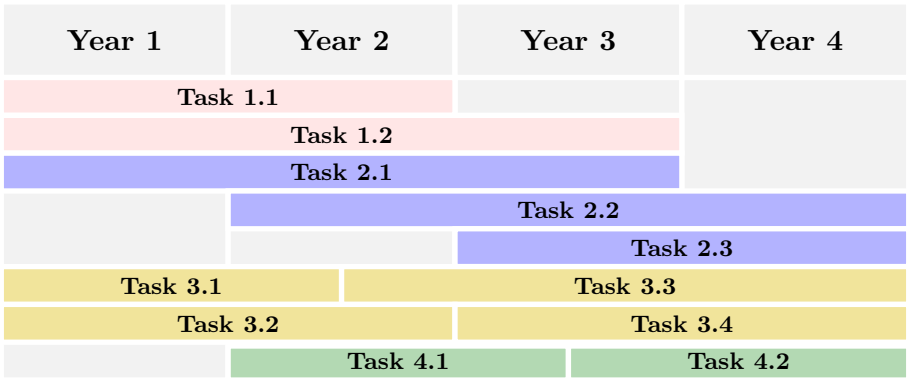
Task 4.2. Analysis of selected e-voting protocols In this subtask we will apply the tools and techniques developed in this project to the protocols identified in subtask 4.1. This will lead in some cases to formal proofs of correctness of the protocols w.r.t. the defined properties and in some cases may identify novel attacks. In any event, we would expect the insights gained from this analysis to suggest novel mechanisms and protocols.

Who does what?

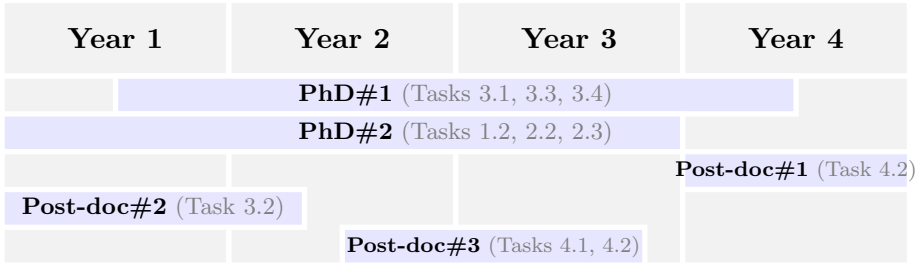
Task	Participants
4.1	INRIA Nancy, Uni Lux.
4.2	INRIA Nancy, Uni Lux.

5.4 Task schedule, deliverables and milestones

Task schedule



Schedule of personal paid by the project



Deliverables and milestones

Task	Date	Title	Person in charge
0	T0+1	website	Steve Kremer
0	T0+12	Progress report	Steve Kremer
0	T0+24	Progress report	Steve Kremer
0	T0+36	Progress report	Steve Kremer
0	T0+48	Final report	Steve Kremer
1	T0+18	Taxonomy of proces equivalences	Peter Ryan
1	T0+36	Composition issues and partial order reductions	David Baelde
2	T0+12	Procedures supporting new features	Steve Kremer
2	T0+18	Improvements of existing tools	David Baelde
2	T0+36	Techniques for verification of an unbounded number of sessions	David Baelde
2	T0+48	Prototype and case studies involving un bounded sessions	David Baelde
3	T0+18	Security definitions and attacker models for weak secrets	Steve Kremer
3	T0+36	Verification techniques for protocols based on weak secrets	Steve Kremer
3	T0+48	Tool prototype and case studies	Steve Kremer
4	T0+18	Security definitions and attacker models for e-voting protocols	Peter Ryan
4	T0+36	Analyses of selected e-voting protocols	Peter Ryan
4	T0+48	Publications based on the case studies	Peter Ryan

5.5 Justification of ressources

5.5.1 Inria Nancy

Equipment

3 laptops 2500 € each **7500 €**

Personnel costs

1 PhD student	36 months	115,200€
1 Post-doc	12 months	48,000€

PhD student: This student will work on Task 3. The aim of this thesis is to design a framework for symbolic verification of protocols relying on low entropy secrets. The student will have to define an extended attacker model, design verification techniques, integrate them into existent tools and validate his work on case studies.

Post-doc: This post-doc will use techniques for verifying equivalence properties and protocols based on low-entropy secrets to formally analyze e-voting protocols that do not require trust in the client platform.

Subcontracting
Nothing.

Travel and meetings

Project meetings				
Project meeting in Cachan	2 meetings	300€/meeting/pers.	3 pers.	1,800 €
Project meeting in Luxembourg	2 meetings	300€/meeting/pers.	3 pers.	1,800 €
Organization of kick-off meeting				350 €
Travel expenses to conferences for dissemination of the results				
International conferences	8 conf.	2,500 €/conf.		20,000 €
Invitation of researchers				
8 visits	1 week	1,000€/week		8,000 €
Total				**31,950.00 €**

Expenses for inward billing (Costs justified by internal procedures of invoicing)
Nothing.

Other working costs
Nothing.

5.5.2 University of Luxembourg

Equipment
Nothing.

Personnel costs

2 Post-docs	15 months each	209,890€ (excluding overheads)

First Post-doc: The post-doc will work on computational definitions for protocols based on low-entropy secrets (Task 3.2). Such protocols include in particular protocols that use empirical out-of-band channels for sending short bitstrings.
Second Post-doc: The post-doc will work on the application of techniques developed in the protocol to e-voting protocols (Task 4.1 and 4.2). He will in particular identify the mechanisms to reduce trust in user platforms and construct formal models for such protocols.

Subcontracting
Nothing.

Travel and meetings

Project meetings				
Project meeting in Nancy	2 meetings	300€/meeting/pers.	2 pers.	1,200 €
Project meeting in Cachan	2 meetings	300€/meeting/pers.	2 pers.	1,200 €
Travel expenses to conferences for dissemination of the results				
International conferences	3 conf.	2,500 €/conf.		7,500 €
Invitation of researchers				
4 visits	1 week	1,000€/week		4,000 €
Total				**13,900 €**

Expenses for inward billing (Costs justified by internal procedures of invoicing)
Nothing.

Other working costs
Nothing.

5.5.3 LSV, ENS Cachan

Equipment
3 laptops 2,000 € each **6,000 €**

Personnel costs
1 PhD student 36 months 97,353€

PhD student: The student will work on Tasks 1 and 2 to design verification techniques that scale better. To achieve this goal, two approaches will be considered. First, concurrency will be taken into account to limit the state space explosion problem that currently hinders trace equivalence checking in security protocols. Second, practical interactive formal verification techniques will be designed to handle unbounded numbers of sessions. Ideally, the two approaches should be combined in a prototype tool developped at the end of the PhD.

Subcontracting
Nothing.

Travel and meetings

Project meetings				
Project meeting in Nancy	2 meetings	300€/meeting/pers.	3 pers.	1,800 €
Project meeting in Luxembourg	2 meetings	300€/meeting/pers.	3 pers.	1,800 €
Organization of final meeting				350 €
Travel expenses to conferences for dissemination of the results				
International conferences	8 conf.	2,500 €/conf.		20,000 €
Invitation of researchers				
8 visits	1 week	1,000€/week		8,000 €
Total				**31,950.00 €**

Expenses for inward billing (Costs justified by internal procedures of invoicing)
Nothing.

Other working costs
Nothing.

6 Valorisation, protection and exploitation of results, global impact of the proposal

Dissemination. We will publish our results in leading journals and in the main conferences in the area of formal methods and security. These conferences include for instance IEEE CSF, IEEE S&P, ACM CCS, POST and ESORICS. Some of the results, e.g. on electronic voting may also be published in more specialized conferences and workshops, such as Vote-ID, EVT/WOTE, ... We believe that this is the best way to ensure international visibility of the project. The website will of course publicize the main results as well.

The tool prototypes produced in the project will be made publicly available on the web and will be open-source, under a free license, such as GPL or CeCILL. This will allow other teams working in security to benefit from this work. Being prototypes in a first stage the tools will mainly be used by colleagues. Again, we believe that the visibility of the project will benefit from making the tools public. We will protect our software by depositing it at APP (Agence pour la Protection des Programmes).

We will also continue our efforts on scientific mediation. Several members of the project have contributed in the recent past to such mediation efforts. These efforts include for instance

- an article in *Pour la Science*:

  R. Chrétien and S. Delaune. La protection des informations sensibles. *Pour La Science* 433, pages 70-77, 2013.

- an article in *Interstices*:

  Vote par Internet, Véronique Cortier, Steve Kremer, 16/01/2013
  `http://interstices.info/vote-internet`

- an article in Inriality:

  Le vote électronique est-il anti-démocratique par essence ? Steve Kremer, Christophe Castro, 29.03.2012.
  `http://www.inriality.fr/vie-citoyenne/elections/securite/le-vote-electronique-est-il/`

as well as participation to the *Fête de la Science* and activities for high school students. We will continue this kind of actions which allow the members of the project to make a broad public aware of the security risks of the numerous digital services that are offered nowadays. In particular, it gives an occasion to inform about the risks and opportunities in both paper and electronic voting schemes.

Impact. Being an academic project the primary impact will be scientific: we will obtain a better understanding of equivalence properties and their usefulness to analyze, find flaws or prove correct protocols which aim to guarantee privacy properties or use low entropy secrets. We believe that, in the long term, this will lead to better protocol design and more secure protocols. In particular, e-voting protocols that do not need to trust the client software are extremely important: the previously described attacks on the real-life Internet voting system used in France and Estonia for parliamentary elections emphasize the need for making progress in this area. Solid theoretical foundations for proving these protocols correct is essential as elections form the cornerstone of modern democracies.

We will also produce tools, which in the short term will be used by other academic groups and semi-academic groups, e.g., Microsoft Research. In the long term our prototypes should evolve towards more mature tools which can be used in industry, as it is for instance the case for AVISPA which is nowadays used by Siemens, SAP,...

References

[3GP10a] 3GPP. Technical specification group core network and terminals; mobile radio interface layer 3 specification; core network protocols; stage 3 (release 9). Technical report, 3rd Generation Partnership Project, 2010. 3GPP TS 24.008 V9.4.0.

[3GP10b] 3GPP. Technical specification group services and system aspects; 3G security; security architecture (release 9). Technical report, 3rd Generation Partnership Project, 2010. 3GPP TS 33.102 V9.3.0.

[3GP11] 3GPP. Technical specification group services and system aspects; 3G security; cryptographic algorithm requirements (release 10). Technical report, 3rd Generation Partnership Project, 2011. 3GPP TS 33.105 V10.0.0.

[ABB+05] A. Armando, D. A. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuéllar, P. H. Drielsma, P.-C. Héam, O. Kouchnarenko, J. Mantovani, S. Mödersheim, D. von Oheimb, M. Rusinowitch, J. Santiago, M. Turuani, L. Viganò, and L. Vigneron. The AVISPA tool for the automated validation of internet security protocols and applications. In *Proc. 17th International Conference on Computer Aided Verification (CAV'05)*, LNCS, pages 281–285. Springer, 2005.

[ACC+08] A. Armando, R. Carbone, L. Compagna, J. Cuellar, and L. T. Abad. Formal analysis of saml 2.0 web browser single sign-on: Breaking the saml-based single sign-on for google apps. In *Proc. 6th ACM Workshop on Formal Methods in Security Engineering (FMSE 2008)*, pages 1–10, 2008.

[ACD12] M. Arapinis, V. Cheval, and S. Delaune. Verifying privacy-type properties in a modular way. In *Proc. 25th IEEE Computer Security Foundations Symposium (CSF'12)*, pages 95–109. IEEE Computer Society Press, 2012.

[ACRR10] M. Arapinis, T. Chothia, E. Ritter, and M. D. Ryan. Analysing unlinkability and anonymity using the applied pi calculus. In *Proc. 23rd Computer Security Foundations Symposium (CSF'10)*, pages 107–121. IEEE Comp. Soc. Press, 2010.

[ADK08] M. Arapinis, S. Delaune, and S. Kremer. From one session to many: Dynamic tags for security protocols. In *Proc. 15th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning (LPAR'08)*, volume 5330 of *LNAI*, pages 128–142. Springer, November 2008.

[AF01] M. Abadi and C. Fournet. Mobile values, new names, and secure communication. In *Proc. 28th ACM Symp. on Principles of Programming Languages (POPL'01)*, pages 104–115. ACM Press, 2001.

[AF04] M. Abadi and C. Fournet. Private authentication. *Theoretical Computer Science*, 322(3):427–476, 2004.

[AG99] M. Abadi and A. D. Gordon. A calculus for cryptographic protocols: The spi calculus. *Information and Computation*, 148(1):1–70, 1999.

[AMR+12] M. Arapinis, L. Mancini, E. Ritter, M. Ryan, N. Golde, K. Redon, and R. Borgaonkar. New privacy issues in mobile telephony: fix and verification. In *Proc. 19th ACM Conference on Computer and Communications Security (CCS'12)*, pages 205–216. ACM Press, 2012.

[And92] J.-M. Andreoli. Logic programming with focusing proofs in linear logic. *J. Log. Comput.*, 2(3):297–347, 1992.

[BAF08] B. Blanchet, M. Abadi, and C. Fournet. Automated verification of selected equivalences for security protocols. *J. Log. Algebr. Program.*, 75(1):3–51, 2008.

[Bau05] M. Baudet. Deciding security of protocols against off-line guessing attacks. In *Proc. 12th ACM Conference on Computer and Communications Security (CCS'05)*, pages 16–25. ACM Press, 2005.

[BCdH10] M. Brusò, K. Chatzikokolakis, and J. den Hartog. Formal verification of privacy for RFID systems. In *Proc. 23rd Computer Security Foundations Symposium (CSF'10)*, pages 75–88. IEEE Comp. Soc. Press, 2010.

[BCL12] G. Bana and H. Comon-Lundh. Towards unconditional soundness: Computationally complete symbolic attacker. In *Proc. 1st International Conference on Principles of Security and Trust (POST'12)*, volume 7215 of *LNCS*, pages 189–208. Springer, 2012.

[BCM12] D. Basin, C. Cremers, and S. Meier. Provably repairing the ISO/IEC 9798 standard for entity authentication. In *Proc. 1st Conference on Principles of Security and Trust (POST'12)*, volume 7215 of *LNCS*, pages 129–148. Springer, 2012.

[BDH14] D. Baelde, S. Delaune, and L. Hirschi. A reduced semantics for deciding trace equivalence using constraint systems. In M. Abadi and S. Kremer, editors, *Proc. 3rd International Conference on Principles of Security and Trust (POST'14)*, LNCS, pages 1–21. Springer, 2014.

[BFG+14] G. Barthe, C. Fournet, B. Grégoire, P.-Y. Strub, N. Swamy, and S. Z. Béguelin. Probabilistic relational verification for cryptographic implementations. In *Proc. 41st Symposium on Principles of Programming Languages (POPL'14)*, pages 193–206. ACM, 2014.

[BGHB11] G. Barthe, B. Grégoire, S. Heraud, and S. Béguelin. Computer-aided security proofs for the working cryptographer. In P. Rogaway, editor, *Advances in Cryptology – CRYPTO 2011*, volume 6841 of *LNCS*, pages 71–90. Springer Berlin Heidelberg, 2011.

[BGS13] M. Backes, M. Gagné, and M. Skoruppa. Using mobile device communication to strengthen e-voting protocols. In *Proceedings of the 12th ACM Workshop on Workshop on Privacy in the Electronic Society*, WPES '13, pages 237–242, New York, NY, USA, 2013. ACM.

[Bla01] B. Blanchet. An Efficient Cryptographic Protocol Verifier Based on Prolog Rules. In *Proc. 14th Computer Security Foundations Workshop (CSFW'01)*, pages 82–96. IEEE Comp. Soc. Press, 2001.

[Bla04] B. Blanchet. Automatic Proof of Strong Secrecy for Security Protocols. In *Proc. Symposium on Security and Privacy (SP'04)*, pages 86–100. IEEE Comp. Soc. Press, 2004.

[BM92] S. M. Bellovin and M. Merritt. Encrypted key exchange: Password-based protocols secure against dictionary attacks. In *Proc. Symposium on Security and Privacy (S&P'92)*, pages 72–84. IEEE Comp. Soc., 1992.

[BPR00] M. Bellare, D. Pointcheval, and P. Rogaway. Authenticated key exchange secure against dictionary attacks. In *In Eurocrypt 2000*, pages 139–155. Springer-Verlag, 2000.

[BPW07] M. Backes, B. Pfitzmann, and M. Waidner. The reactive simulatability (RSIM) framework for asynchronous systems. *Information and Computation*, 205(12):1685–1720, 2007.

[BR94] M. Bellare and P. Rogaway. Entity authentication and key distribution. In *Proc. 13th Annual International Cryptology Conference on Advances in Cryptology (CRYPTO '93)*, pages 232–249. Springer, 1994.

[BSSW02] D. Balfanz, D. K. Smetters, P. Stewart, and H. C. Wong. Talking to strangers: Authentication in ad-hoc wireless networks. In *Proc. Network and Distributed System Security Symposium (NDSS'02)*. The Internet Society, 2002.

[BU13] F. Böhl and D. Unruh. Symbolic universal composability. In *Proc. 26th Computer Security Foundations Symposium (CSF'13)*, pages 257–271. IEEE Comp. Soc. Press, 2013.

[Can01] R. Canetti. Universally composable security: A new paradigm for cryptographic protocols. In M. Naor, editor, *Proc. 42nd IEEE Symp. on Foundations of Computer Science (FOCS'01)*, pages 136–145. IEEE Comp. Soc. Press, 2001.

[CB13] V. Cheval and B. Blanchet. Proving more observational equivalences with proverif. In *Proc. International Conference on Principles of Security and Trust (POST'13*, volume 7796 of *LNCS*, pages 226–246. Springer, 2013.

[CC10] Ş. Ciobâcă and V. Cortier. Protocol composition for arbitrary primitives. In *Proc. 23rd IEEE Computer Security Foundations Symposium (CSF'10)*, pages 322–336. IEEE Comp. Soc. Press, 2010.

[CCD11] V. Cheval, H. Comon-Lundh, and S. Delaune. Trace equivalence decision: Negative tests and non-determinism. In *Proc. 18th ACM Conference on Computer and Communications Security (CCS'11)*, pages 321–330. ACM Press, 2011.

[CCD13] R. Chrétien, V. Cortier, and S. Delaune. From security protocols to pushdown automata. In *Proc. 40th International Colloquium on Automata, Languages and Programming (ICALP'13)*, volume 7966 of *LNCS*, pages 137–149. Springer, 2013.

[CCK12] R. Chadha, Ş. Ciobâcă, and S. Kremer. Automated verification of equivalence properties of cryptographic protocols. In *Proc. 21th European Symposium on Programming (ESOP'12)*, volume 7211 of *LNCS*, pages 108–127. Springer, 2012.

[CD09] V. Cortier and S. Delaune. Safely composing security protocols. *Formal Methods in System Design*, 34(1):1–36, February 2009.

[CDKR13] C. Chevalier, S. Delaune, S. Kremer, and M. D. Ryan. Composition of password-based protocols. *Formal Methods in System Design*, 43:369–413, 2013.

[CDL06] V. Cortier, S. Delaune, and P. Lafourcade. A survey of algebraic properties used in cryptographic protocols. *Journal of Computer Security*, 14(1):1–43, 2006.

[Cha01] D. Chaum. Surevote: Technical overview. In *Proc. Workshop on Trustworthy Elections (WOTE'01)*, 2001.

[Che14] V. Cheval. Apte: an algorithm for proving trace equivalence. In *Proc. 20th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'14)*, volume 8413 of *LNCS*. Springer, 2014.

[CHK+05] R. Canetti, S. Halevi, J. Katz, Y. Lindell, and P. Mackenzie. Universally composable password-based key exchange. In *Advances in Cryptology - Eurocrypt 2005, LNCS*, pages 404–421. Springer-Verlag, 2005.

[CLCS13] H. Comon-Lundh, V. Cortier, and G. Scerri. Tractable inference systems: An extension with a deducibility predicate. In *Proic. 24th International Conference on Automated Deduction (CADE'13)*, volume 7898 of *LNCS*, pages 91–108. Springer, 2013.

[Cre08] C. Cremers. The Scyther Tool: Verification, falsification, and analysis of security protocols. In *Proc. 20th International Conference on Computer Aided Verification (CAV'08)*, volume 5123 of *LNCS*, pages 414–418. Springer, 2008.

[Cre11] C. Cremers. Examining indistinguishability-based security models for key exchange protocols: The case of ck, ck-hmqv, and eck. In *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*, ASIACCS '11, pages 80–91, New York, NY, USA, 2011. ACM.

[CS10] T. Chothia and V. Smirnov. A traceability attack against e-passports. In *Proc. 14th International Conference on Financial Cryptography and Data Security (FC'10)*, volume 6052 of *LNCS*, pages 20–34. Springer, 2010.

[DDS10] M. Dahl, S. Delaune, and G. Steel. Formal analysis of privacy for vehicular mix-zones. In *Proc. 15th European Symposium on Research in Computer Security (ESORICS'10)*, volume 6345 of *LNCS*, pages 55–70. Springer, 2010.

[DJP10] N. Dong, H. Jonker, and J. Pang. Analysis of a receipt-free auction protocol in the applied pi calculus. In S. Etalle and J. Guttman, editors, *Proc. International Workshop on Formal Aspects in Security and Trust (FAST'10)*, 2010.

[DKP09] S. Delaune, S. Kremer, and O. Pereira. Simulation based security in the applied pi calculus. In *Proceedings of the 29th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'09)*, volume 4 of *Leibniz International Proceedings in Informatics*, pages 169–180. Leibniz-Zentrum für Informatik, 2009.

[DKR09] S. Delaune, S. Kremer, and M. D. Ryan. Verifying privacy-type properties of electronic voting protocols. *Journal of Computer Security*, 17(4):435–487, July 2009.

[DLM04] N. A. Durgin, P. Lincoln, and J. C. Mitchell. Multiset rewriting and the complexity of bounded security protocols. *Journal of Computer Security*, 12(2):247–311, 2004.

[EMM09] S. Escobar, C. Meadows, and J. Meseguer. Maude-NPA: Cryptographic protocol analysis modulo equational properties. In *Foundations of Security Analysis and Design V*, volume 5705 of *LNCS*, pages 1–50. Springer, 2009.

[For04] P. T. Force. PKI for machine readable travel documents offering ICC read-only access. Technical report, International Civil Aviation Organization, 2004.

[GN04] C. Gehrmann and K. Nyberg. Manual authentication for wireless devices. *RSA Cryptobytes*, 7:2004, 2004.

[GRCC] G. Grewal, M. Ryan, L. Chen, and M. Clarkson. Du-vote: Remote voting with untrusted computers. Unpublished manuscript.

[GT00] J. D. Guttman and F. J. Thayer. Protocol independence through disjoint encryption. In *Proc. 13th Computer Security Foundations Workshop (CSFW'00)*, pages 24–34. IEEE Comp. Soc. Press, 2000.

[HK14] R. Haenni and R. Koenig. *Design, Development, and Use of Secure Electronic Voting Systems*, chapter Voting over the Internet on an Insecure Platform. IGI Global, mar 2014.

[Hoe05] J.-H. Hoepman. Ephemeral pairing on anonymous networks. In *Proc. 2nd International Conference on Security in Pervasive Computing (SPC'05)*, LNCS, pages 101–116. Springer, 2005.

[JFR13] R. Joaquim, P. Ferreira, and C. Ribeiro. Eviv: An end-to-end verifiable internet voting system. *Computers & Security*, 32(0):170 – 191, 2013.

[Joa14] R. Joaquim. How to prove the validity of a complex ballot encryption to the voter and the public. *Journal of Information Security and Applications*, 2014. To appear.

[JRF09] R. Joaquim, C. Ribeiro, and P. Ferreira. Veryvote: A voter verifiable code voting system. In P. Ryan and B. Schoenmakers, editors, *VOTE-ID 2009*, volume 5767 of *LNCS*, pages 106–121, Luxembourg, September 2009. Springer.

[KLH13] R. Koenig, P. Locher, and R. Haenni. Attacking the verification code mechanism in the norwegian internet voting system. In *E-Voting and Identify*, volume 7985 of *LNCS*, pages 76–92. Springer, 2013.

[KOY01] J. Katz, R. Ostrovsky, and M. Yung. Efficient password-authenticated key exchange using human-memorable passwords. In *Proc. International Conference on the Theory and Application of Cryptographic Techniques: Advances in Cryptology (EUROCRYPT'01)*, pages 475–494. Springer, 2001.

[KR05] S. Kremer and M. D. Ryan. Analysis of an electronic voting protocol in the applied pi-calculus. In *Programming Languages and Systems — Proceedings of the 14th European Symposium on Programming (ESOP'05)*, volume 3444 of *LNCS*, pages 186–200. Springer, 2005.

[KV13] J. Katz and V. Vaikuntanathan. Round-optimal password-based authenticated key exchange. *Journal of cryptology*, 26(4):714–743, 2013.

[Low95] G. Lowe. An attack on the Needham-Schroeder public key authentication protocol. *Information Processing Letters*, 56(3):131–133, 1995.

[Low97] G. Lowe. Casper: a compiler for the analysis of security protocols. In *Proc. 10th Computer Security Foundations Workshop (CSFW'97)*, pages 18–30. IEEE Computer Society Press, 1997.

[Low03] G. Lowe. Analysing protocols subject to guessing attacks. In *Journal of Computer Security*, page 2004, 2003.

[MVB10] S. Mödersheim, L. Viganò, and D. A. Basin. Constraint differentiation: Search-space reduction for the constraint-based analysis of security protocols. *Journal of Computer Security*, 18(4):575–618, 2010.

[Nef04] C. A. Neff. Practical high certainty intent verification for encrypted votes, 2004.

[NR11] L. Nguyen and A. W. Roscoe. Authentication protocols based on low-bandwidth unspoofable channels: A comparative survey. *Journal of Computer Security*, 19(1):139–201, 2011.

[Pel98] D. Peled. Ten years of partial order reduction. In *Proc. 10th International Conference on Computer Aided Verification, CAV'98*, volume 1427 of *LNCS*. Springer, 1998.

[Poi12] D. Pointcheval. Password-based authenticated key exchange. In *Public Key Cryptography–PKC 2012*, pages 390–397. Springer, 2012.

[Ros94] A. W. Roscoe. *Essays in Honour of C. A. R. Hoare*, chapter Model-checking CSP. Prentice Hall, 1994.

[RS99] P. Ryan and S. Schneider. Process algebra and non-interference. In *Proc. 12th IEEE IEEE Computer Security Foundations Workshop (CSFW'99)*, pages 214–227. IEEE Comp. Soc. Press, 1999.

[RSG+00] P. Ryan, S. Schneider, M. Goldsmith, G. Lowe, and A. Roscoe. *Modelling and Analysis of Security Protocols*. Addison Wesley, 2000.

[RSN11] A. Roscoe, T. Smyth, and L. Nguyen. Model checking cryptographic protocols subject to combinatorial attack. Technical report, Oxford University, 2011.

[RT01] M. Rusinowitch and M. Turuani. Protocol insecurity with finite number of sessions is NP-complete. In *Proc. 14th Computer Security Foundations Workshop (CSFW'01)*, pages 174–190. IEEE Comp. Soc. Press, 2001.

[RT09] P. Ryan and V. Teague. Pretty good democracy. In *Proc. 17th Security Protocols Workshop*, LNCS. Springer, 2009.

[RW06] R. L. Rivest and J. P. Wack. On the notion of "software independence" in voting systems. Prepared for the TGDC, and available at `http://people.csail.mit.edu/rivest/RivestWack-OnTheNotionOfSoftwareIndependenceInVotingSystems.pdf`, 2006.

[RWW94] A. W. Roscoe, J. C. P. Woodcock, and L. Wulf. Non-interference through determinism. In *Proc. 3rd European Symposium on Research in Computer Security (ESORICS'94)*, pages 33–53. Springer, 1994.

[Rya01] P. Y. A. Ryan. Mathematical models of computer security. In *Revised Versions of Lectures Given During the IFIP WG 1.7 International School on Foundations of Security Analysis and Design on Foundations of Security Analysis and Design: Tutorial Lectures (FOSAD'00)*, pages 1–62. Springer, 2001.

[SMCB12] B. Schmidt, S. Meier, C. Cremers, and D. Basin. Automated analysis of Diffie-Hellman protocols and advanced security properties. In *Proc. 25th IEEE Computer Security Foundations Symposium (CSF'12)*, pages 78–94. IEEE Comp. Soc. Press, 2012.

[TD10] A. Tiu and J. Dawson. Automating open bisimulation checking for the spi-calculus. In *Proc. 23rd Computer Security Foundations Symposium (CSF'10)*, pages 307–321. IEEE Comp. Soc. Press, 2010.

[Tur06] M. Turuani. The CL-AtSe protocol analyser. In *Proc. 17th International Conference on Rewriting Techniques and Applications (RTA'06)*, volume 4098 of *LNCS*, pages 277–286. Springer, 2006.

[Vau05] S. Vaudenay. Secure communications over insecure channels based on short authenticated strings. In *Proc. 25th Annual International Cryptology Conference on Advances in Cryptology (CRYPTO '05)*, pages 309–326. Springer, 2005.

[vDR08] T. van Deursen and S. Radomirovic. Attacks on RFID protocols. Cryptology ePrint Archive, Report 2008/310, 2008. `http://eprint.iacr.org/`.