5-minute Talks

1. *Privacy-supporting cloud-based conference management*

   Myrto Arapinis, Sergiu Bursuc, Mark D Ryan

   We describe the protocol underlying a novel cloud-based conference management system that offers strong security and privacy properties. Authors, reviewers and the conference chair interact through their web browsers with the cloud, to perform the usual tasks of uploading and downloading papers and reviews. In contrast with current systems, the cloud provider does not have access to the content of papers and reviews, and moreover is unable to link authors, reviewers, and scores (for example, even though it knows Alice and Bob, it cannot deduce that Alice reviewed a paper by Bob). We provide a prototype implementation of the system and performance results. We also express the protocol and its desired privacy properties in the language of ProVerif, and automatically prove that the properties hold.

   The talk implements some ideas motivated in the CACM opinion piece at `http://cacm.acm.org/opinion/articles/103200-cloud-computing-privacy-concerns-on-our-doorstep/pdf`

2. *Observational equivalence and Labelled Bisimilarity in the Applied Pi-Calculus revisited*

   Myrto Arapinis, Eike Ritter and Mark Ryan

   The applied pi-calculus is an extension of the pi-calculus with terms and equations. It has been used extensively to specify and verify properties of security protocols. One of the key theorems for the applied pi-calculus is that observational equivalence and labelled bisimulation coincide. This theorem is important because observational equivalence is used to specify security properties, and labelled bisimulation is used to verify them. This theorem was stated but not proved in the original paper by Abadi and Fournet in 2001. A counterexample was found in 2010 by Bengtson/Johansson/Parrow/Victor. Rather than introducing a new calculus, like BJPV, we only change the definition of the operational semantics and hence of the labelled bisimulation in a suitable way but otherwise keep the applied pi-calculus and the observational equivalence. This is sufficient to obtain the above theorem.

3. *Automatic Verification of Group Protocols with Unbounded Numbers of Participants and Sessions*

   Miriam Paiola and Bruno Blanchet

   We present a novel automatic technique for proving secrecy properties of group protocols for an unbounded number of participants and an unbounded number of sessions. This result is achieved by extending the Horn clause approach of the automatic protocol verifier ProVerif. We extend the Horn clauses to be able to represent messages that depend on the number of participants. We adapt the resolution algorithm to handle the new class of Horn clauses, and prove the soundness of this new algorithm. We illustrate our algorithm on a version of the Asokan-Ginzboorg protocol.

4. *A decision procedure for trace equivalence*

   Vincent Cheval, Hubert Comon-Lundh, and Stphanie Delaune.

   Security protocols are used today to secure transaction that rely on public channels like the Internet. It is therefore essential to obtain as much confidence as possible in their correctness. Many works have been devoted to the use of formal methods to analyse the security of these protocols. Many tools have also been developed to automatically verify cryptographic protocols (e.g. ProVerif).

   Until recently, most efforts and successes only concerned trace properties, i.e. security properties that can be checked on each individual sequence of messages corresponding to an execution of the protocol. Secrecy and authentication are typical examples of trace properties. There are however several security properties, which cannot be defined as trace properties and require the notion of equivalence. We focus here in automating the proofs of trace equivalence which is well-suited for the analysis of security protocols.

A line of works consists in designing stronger notions of equivalences that imply observational equivalence (and thus trace equivalence). ProVerif implements an algorithm, based on Horn clauses and dedicated resolution strategies, which is able to establish the observational equivalence between two processes written in the applied pi calculus. However, all these methods check a stronger equivalence than observational equivalence and fail on some simple toy examples. Unfortunately, this is exactly the kind of situations we encountered in several case studies, e.g. the private authentication protocol, and e-passport protocols. In summary, there is no known result that is suitable to the two previous protocols : these protocols require a conditional (with a non-trivial else branch) to be modeled in accurate way. Moreover, the notion of equivalence used by ProVerif is too strong to conclude on these case studies and yields a false negative.

Our main contribution consists in providing with a new algorithm, that decides the trace equivalence of (possibly non-determinate, with non-trivial else branches) processes, without replication and that use standard primitives, namely signature, hash function, pairing, symmetric and asymmetric encryptions.

5. *Logical Protocol Analysis for Authenticated Diffie-Hellman*

   Daniel J. Dougherty and Joshua D. Guttman

   Diffie-Hellman protocols for authenticated key agreement construct a shared secret with a peer using a minimum of communication and using limited cryptographic operations. However, their analysis has been challenging in computational models and especially in symbolic models.

   In this paper, we develop a framework for protocol analysis that combines algebraic and strand space ideas. We show that it identifies exact assumptions on the behavior of a certifying authority. These assumptions establish the confidentiality and authentication properties for two protocols, the Unified Model and Menezes-Qu-Vanstone (MQV). For MQV, we establish a stronger authentication property than previously claimed, using a stronger (but realistic) assumption on the certifying authority.

   Verification within our framework implies that the adversary has no strategy that works uniformly, independent of the choice of the cyclic group in which the protocol operates. Indeed, we provide an equational theory which constitutes an analysis of these uniform strategies. We provide an abstraction, the notion of indicator, which leads to easy proofs of protocol correctness assertions. Computational soundness awaits further investigation.

6. *Translating between equational theories for automated reasoning (work in progress)*

   Ben Smyth

   In the symbolic model of security protocol analysis, cryptographic primitives are modelled as function symbols and the relationships between these primitives are captured using equational theories. Unfortunately, the desired set of function symbols and associated equations may not be amenable to automated analysis; for example, the equational theory may be non-convergent, non-linear, or the set of function symbols may be infinite. In practice, this problem can sometimes be overcome by translating data terms to a different equational theory. In this talk, we propose some conditions for translation between equational theories that aim to preserve observational equivalence properties. Our work in progress aims to show that these rules are indeed sound.

7. *Computational soundness of symmetric encryption in the presence of malicious keys (work in progress)*

   Michael Backes and Esfandiar Mohammadi

   Security proofs of cryptographic protocols are lengthy and error-prone. Consequently, there is a long and promising history of attempts to automatize the veri

   cation of these protocols. Due to its complexity, typically symbolic protocols have been analysed in which cryptographic operations are abstracted as symbolic operations that obey cancellation rules and the symbolic attacker is restricted to these symbolic operations (e.g., $\mathsf{dec}(\mathsf{enc}(m,k),k) = m$). Of course, an analysis of such a symbolic abstraction (called a Dolev-Yao model) does not yield per se security guarantees against a real-world attacker. This gap has been bridged by a sequence of so-called computational soundness results, proving that the verification of a Dolev-Yao model yield security guarantees against a cryptographic attacker.

Dolev-Yao models comprising symmetric encryption have been widely analysed and studied. But so far no comprehensive result for the computational soundness of symmetric encryption has been presented in the standard model for the following reason: As the security of a symmetric encryption (say IND-CCA) only considers honestly generated keys, the injection of malicious keys can cause unpredictable protocol behavior. As an example assume a scenario in which the attacker already sent a ciphertext; then the adversary can find a key that decrypts to a value that was not even known to the attacker at the time the ciphertext was sent. For public-key encryption such a problem does not arise since the ciphertext can be tagged with the public-key, which in turn can act as a commitment to the secret key.

In this work, we consider as a symbolic (Dolev-Yao) model that comprises symmetric encryption and allows protocols in which the attacker may inject dishonestly generated keys. For this symbolic model, we present a computational soundness result for observational equivalence properties in the standard model. The main challenge are the cases in which the attacker sends a ciphertext and later sends a key for that ciphertext. We address this challenge by requiring each ciphertext to be tagged with a commitment on the key and the message. This commitment is extractable by a quasi-poly simulator (i.e., with running time nlog n) and, hence, allows the simulator to determine one (symbolically) valid key for attacker-generated ciphertext.

8. *A soundness result for dishonest keys*

Hubert Comon-Lundh, Veronique Vortier, Guillaume Scerri

Several formal models for protocol verification have been introduced in the last 30 years. These models allow the development of automatic security proofs, and help in the task of finding attacks. The most famous case is the attack (and fix) on the Needham-Schroeder protocol discovered by G. Lowe, and now the Needham-Schroeder-Lowe (NSL) protocol is a benchmark for all demonstration tools. But even if the NSL protocol has been proven hundreds of times, it is only secure in the abstract model. For example, if an El-Gamal cryptosystem is used for implementing the asymmetric encryption scheme, B. Warinschi found an attack for some reasonable implementation of the pair.

This paradox is due to the fact that abstract models are too abstract, in the sense that cryptographic primitives are idealized, and implementation hypotheses are undefined. This is why in the last ten years there have been lots of results dealing with that problem. More precisely results saying that if there are attacks on a protocol in a concrete model (for example Canetti's), these attacks should also exist in the abstract model, which means that the abstract model should be sound with respect to the concrete model. The first result of this type is due to M. Abadi and P. Rogaway in 2000 and is limited to a passive attacker. Several generalisations of this result do exist, for example one by D. Micciancio and B. Warinschi in 2004, and one by M. Backes, D. Hofheinz and D. Unruh in 2009.

All results capturing encryption rely on the hypothesis that no dishonest keys are used by the attacker, meaning that each key comes with a certificate. If this hypothesis can be reasonable in the case of asymmetric encryption, it is unrealistic in the case of symmetric encryption. This hypothesis is necessary in the existing models, because cryptographic hypotheses allow dishonest keys to have arbitrary properties. An adversary using dishonest keys can, for example, create arbitrary equalities between decryptions with this dishonest key and deducible terms, as the decryption by a dishonest key can be an arbitrary function. He can also learn one bit of information on a term t each time a honest agent tries to decrypt it with a dishonest key, as the success of the decryption can be used as an oracle for the nth bit. As existing abstract models do not provide any mean to add equalities or model hidden information flows, there is a need to design a new abstract model in order to have soundness in presence of dishonest keys.

Our contribution is to provide a model allowing to increased the adversary's capabilities due to the usage of dishonest keys, which is sound with respect to the computational model for both trace and equivalence properties, under reasonable cryptographic hypotheses. This work is inspired by the work of H. Comon and V. Cortier in 2008 proving soundness of observational equivalence. The originality of our approach consists in allowing the symbolic attacker to create new equalities that soundly reflect the behaviour of dishonest keys.

9. *Computational soundness with unrestricted key use*

   Michael Backes, Ankit Malik, and Dominique Unruh

   Computational soundness results are theorems showing that protocols secure in a symbolic (Dolev-Yao) model of cryptography are also secure in a computational model of cryptography. This allows to get the best of two worlds: The simplicity of symbolic cryptography, combined with the realistic attacker model from computational cryptography.

   State-of-the-art computational soundness results for public key encryption do, however, impose strong limitations on the use of secret keys. Protocols in which honest parties transmit secret keys, receive secret keys, or construct key-cycles involving the secret keys are explicitly excluded. This even holds if on the symbolic side, these operations do not threaten the security of the protocol.

   We present a computational soundness result that permits sending and receiving of secret keys, even in the presence of key cycles in the random oracle model. For this, we need to solve several problems:

   - Receiving secret keys: The standard approach for proving computational soundness is to construct a simulator that translates incoming messages into terms (parsing). If a secret key is received too late, this fails. We solve this by using "lazy parsing". When the simulator encounters an encryption he cannot parse, he can delay parsing until needed.
   - Sending secret keys: Here, we run into the key commitment problem: In the proof of computational soundness, one typically replaces all encryptions with respect to uncorrupted keys by fake encryptions to 0. This leads to problems when the secret key is later sent to the adversary because then the adversary notices the difference. We solve this by introducing a new security notion for encryption schemes in the random oracle model: Programmable key-dependent-message security (PROG-KDM). A PROG-KDM secure scheme allows us to "patch" the random oracle to retroactively make an existing fake encryption into an encryption of an arbitrary plaintext. PROG-KDM security can be achieved by standard constructions in the random oracle model and even handles key-cycles.

10. *Reasoning about differentially private computations*

    Gilles Barthe, Boris Koepf, Federico Olmedo, and Santiago Zanella

    We report on CertiPriv, a machine-checked framework to prove differential privacy of programs from first principles. CertiPriv enables one to formally certify (i) the correctness of existing mechanisms for differential privacy, such as the Laplacian and Exponential mechanism, (ii) the differential privacy of algorithms with arbitrary (not necessarily numeric) output domains, and (iii) the differential privacy of complex interleavings of (not necessarily differentially private) probabilistic computations.

11. *Title Policy Auditing over Incomplete Logs*

    Deepak Garg, Limin Jia, and Anupam Datta

    We present the design, implementation and evaluation of an algorithm that checks audit logs for compliance with privacy and security policies. The algorithm, which we name reduce, addresses two fundamental challenges in compliance checking that arise in practice. First, in order to be applicable to realistic policies, reduce operates on policies expressed in a first-order logic that allows restricted quantification over infifinite domains. We build on ideas from logic programming to identify the restricted form of quantified formulas. The logic can, in particular, express all 84 disclosure-related clauses of the HIPAA Privacy Rule, which involve quantification over the infinite set of messages containing personal information. Second, since audit logs are inherently incom- plete (they may not contain sufficient information to determine whether a policy is violated or not), reduce proceeds iteratively: in each iteration, it provably checks as much of the policy as possible over the current log and outputs a residual policy that can only be checked when the log is extended with additional information. We prove correctness, termination, time and space complexity results for reduce. We implement reduce and optimize the base implementation using two heuristics for database indexing that are guided by the syntactic structure of policies. The implementation is used to check simulated audit logs for compliance with the HIPAA Privacy Rule. Our experimental results demonstrate that the algorithm is fast enough to be used in practice.

12. *Security and Trust Foundations (STF)*

    Joshua Guttman

    STF is the new, sixth main ETAPS conference that will have its first meeting in ETAPS 2012 in Tallinn next March.

    Security and Trust Foundations is a broad forum related to the theoretical and foundational aspects of security and trust. Papers of many kinds are welcomed: new theoretical results, practical applications of existing foundational ideas, and innovative theoretical approaches stimulated by pressing practical problems.

    Abstracts will be due 7 October, and papers on 14 October.

13. *Tracking 3G mobile phone users*

    Myrto Arapinis, L. I. Mancini, Eike Ritter, and Mark Ryan

    Universal Mobile Telecommunications System (UMTS) is a mobile telephony standard specified and maintained by the Third Generation Partnership Project (3GPP). UMTS was introduced in 1999 to offer a better support for mobile data applications increasing the data rate and lowering the costs of mobile data communications. Furthermore, UMTS offers an improved security architecture with respect to previous mobile communication systems such as GSM (Global System for Mobile Communication). In particular, the UMTS communication system aims to provide user untraceability, i.e. aims to make it impossible to any outside observer to deduce whether different services are delivered to the same user.

    To avoid untraceability, a mobile station uses a temporary identity (TMSI Temporary Mobile Subscriber Identity) to identify itself to the network, instead of using its fixed unique identity (IMSI International Mobile Subscriber Identity). Moreover, the UMTS standard requires the mobile station performs periodic updates of its temporary identity by executing the TMSI reallocation procedure.

    In this talk, we will show that the location update procedure is awed in a way that makes it possible to trace the movements of a particular mobile station, and thus of its bearer. More precisely, it is the Authentication and Key Agreement (AKA) protocol, on which relies the location update procedure, that leeks information on the identity of the mobile station. We will see that all an attacker has to do is to sniff one session between the mobile device and the legitimate network and record a particular message. Then by replaying the recorded message, the attacker can distinguish that mobile device from any other.

14. *Small TCBs for policy-controlled operating systems*

    Winfried E. Khnhauser and Anja Plck

    Unlike most software systems, Trusted Computing Bases (TCBs) of today's commodity systems have not been designed by methodical software engineering: they have evolved for decades, with independent contributions from many people and organizations. As a result, TCB implementations are heterogeneous, large, redundant, distributed, and complex, rendering vital TCB properties such as correctness, robustness, and tamperproofness difficult to assert.

    Recently, new kernel abstractions providing a runtime environment for security policies together with policy enforcement mechanisms have been integrated into contemporary operating systems. Security policy specifications are now integrated into the OS kernel, so that the TCB implementation is less distributed, less inhomogeneous, and easier to identify and isolate, but still neither smaller nor less complex. The complexity originates from the ambition to support a wide variety of policies. Hence, the kernel's policy runtime support is designed to pursue generality, rendering the TCB still large, complex, and expensive.

    We follow a different approach: the idea is to design policy-specific TCBs where the extent of policy support is tailored to the policies that are present in a TCB. In order to identify the functions of a policy's execution environment, we exploit causal dependencies between policies and TCB functions. This results in causally determined TCBs containing only those functions which are both necessary and sufficient to enforce and protect the TCBs' security policies.