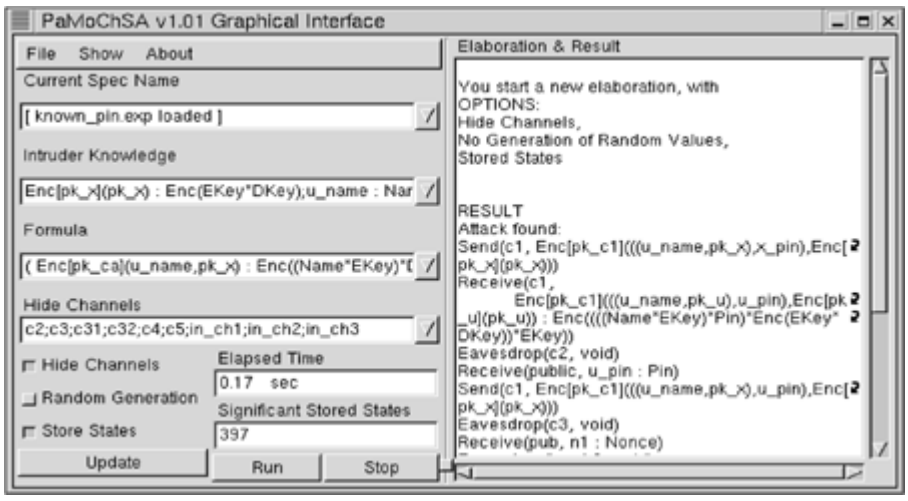
Security: Formal Methods at Work

by Stefano Bistarelli, Fabio Martinelli and Marinella Petrocchi

Security is becoming a crucial issue in economic and social activities that involve electronic transactions. The 'Istituto di Informatica e Telematica' (IIT-CNR) is conducting several activities in the field of computer security. In particular, one group is involved in the definition and application of correct and rigorous formal methods for the analysis of network and system security aspects.

Cryptography has long been regarded as the main practical means to protect the confidentiality of information traveling on the communication networks. It is now also being adopted in many more complex applications, where the correctness of the algorithm does not guarantee 'per se' the correctness of the application. Procedures that apply cryptography are largely being used at the moment for message authentication, personal identification, digital signatures, electronic money transfer and other critical applications. Even if we assume that the cryptography in such procedures is completely reliable, weaknesses may result from the way in which it is used and assembled in the communication protocol. Noteworthy examples of this range from academic cryptographic protocols, such as the Needham-Schroeder public key protocol (1978), which was believed to be correct for several years until shown to be flawed by Lowe in 1996 (using formal techniques), to industrial applications, such as the Java programming language (which was found to have type flaws leading to security holes) and the recently announced security holes in Netscape Navigator and Internet Explorer. Many of these could conceivably have been prevented by a careful formal design and analysis.



PaMoChSA graphical interface.

The detection and prevention of bugs are in fact two of the main reasons for using formal methods and related approaches: the specification of a system is an essential tool for analysis, and may help to discover many design errors. If the specification is given in an executable language, system execution can be simulated, making it easier to verify certain properties (early prototyping). Other reasons to use formal specifications typically include the need to express user requirements unambiguously, and to produce a reference guide for the system implementer.

In the formal analysis approach, a security protocol (or architecture) is commonly described as a process in an executable specification language. This process is designed to act in a hostile environment, usually represented as another process of the language (the attacker). In the worst-case analysis scenario, the attacker has complete control over the communication network, ie, it can intercept, fake and eavesdrop all communications. The entire system can be analyzed by applying specific techniques. For instance, security is sometimes analyzed by comparing the state-space resulting from the execution of the protocol with and without the attacker. The differences may represent possible attacks that have to be carefully studied. It is worth noting that the attacker is able to deduce new messages from the messages it has received during a computation. The basic algebraic features of cryptographic functions are represented as rewriting rules for terms of a language that denote cryptographic messages. This means that there may be rules that allow an attacker to discover a message encrypted with a certain key when the attacker also holds the correct decryption key.

Analysis methods of this type can be also implemented in automated software tools. These tools can be used by (reasonably) non-expert people and, hopefully, by the end-user of a security application in order to achieve a better comprehension of the security mechanisms offered by the application itself.

Our current and future activities in the field of formal analysis of computer security can be summarized as follows:

Theoretical: Our goal is to develop new and more efficient analysis techniques for security protocols and open systems. Recent advances concern the simulation of possible attacks using symbolic techniques to represent the state-space of the system under attack more succinctly. Other techniques aim at defining quality measures with respect to the relevance of possible attacks on security protocols, by enabling assessment of the relative merits of the protocols.

Applicative: We are now developing and testing a software tool (PaMoChSA, or Partial Model Checking Security Analyzer) implementing our analysis techniques. Features of the current implementation include:

- possibility to check a number of security properties, eg confidentiality, message and entity authentication, integrity
- no specification needed for the attacker
- the underlying theory is almost parametric with respect to the set of term rewriting rules for modeling cryptography
- a compiler translating from the common (and ambiguous) notation for security protocols used in the literature to a more accurate notation based on formal description techniques.

We are now applying our verification tool to real-life case studies. For example, we have performed a conceptual analysis of some procedures of the open source software OpenCA, which is basically a set of procedures for running a Certification Authority, issuing X.509 digital certificates. We have also analyzed some security mechanisms of the Simple Certificate Enrollment Protocol (SCEP).

Several national agencies and institutions support our research, for instance the Italian National Research Council (CNR), the Italian Ministry for the University and Scientific and Technological Research (MURST), the Center of Excellence for Research, Development and Demonstration of Advanced Information and Communication Technology (CSP).

Link:

http://www.iat.cnr.it/attivita/progetti/progetti.html

Please contact:

Fabio Martinelli, IIT-CNR

Tel: +39 050 315 3425

E-mail: fabio.martinelli@iit.cnr.it