
Thesis description:

Many cryptographic protocols, such as non-repudiation, contract-signing, voting, auction, exams, involve the use of third parties, such as notaries, judges, or authorities. It is crucial that such parties can be held accountable in case they misbehave as this is a strong incentive for such parties to follow the protocol. In security protocol design the notions of Verifiability and Accountability [1] have been known for some time. A verifiable protocol allows the parties to verify, after the protocols execution, whether the outcome of the protocol is correct. When verifiability fails, accountability additionally allows to identify which party is responsible for this failure.

Currently, there are automatic protocol verifiers (e.g. ProVerif [http://proverif.inria.fr]) that permit the analysis of accountability in security protocols that involve at least one semi-trusted party. Normally, they cannot consider fully corrupted parties because they implement an attacker model (known as Dolev-Yao attacker [2]) that makes very difficult to check security properties without resorting on some trust assumption.

A new threat model known as Multi-Attacker [3] has been advanced more recently. It allows every party to be corrupted, however each of them neither colludes nor shares knowledge with any other party. Hence, the multi-attacker ideally permits to analyse security protocols assuming fully corrupted parties with conflicting goals.

The goal of this project is to implement the Multi-Attacker threat model in an automatic protocol verifier and check accountability definitions on a case study protocol considering no trusted parties.


Tasks:

- Identify and analyse relevant related work
- Propose an implementation of the Multi-Attacker in a protocol verifier of choice
- Choose a case study protocol and formalise it in the verifier
- Evaluate the results

Competence:

- Knowledge of functional programming
- Knowledge of the OCaml language would be a plus. (The knowledge of the object part of OCaml would be not required.)
- Knowledge of process calculae, such as the pi calculus or other similar calculae, and of first-order logic would be a plus.

Applications should include a brief personal letter, CV, and recent grades. In your application, make sure to give examples of previous programming or other projects that you consider relevant for the position.

Expected start time: As soon as possible

Location: SICS Swedish ICT AB, Lund

Contact person/s: Dr. Rosario Giustolisi (rosario.giustolisi@sics.se)
About SICS
SICS Swedish ICT is a leading research institute for applied information and communication technology in Sweden. SICS is a part of Swedish ICT Research AB, a non-profit research organization owned by the Swedish government and industry. SICS’ mission is to contribute to the competitive strength of Swedish industry by conducting advanced and focused research in strategic areas of computer science, and actively promote the uptake of new research ideas and results in industry and society at large. SICS is an active participant in collaborative national, European, and other international R&D programs.
SICS Security lab main research areas are Cloud Security, Security in the Internet of Things (IoT) and Secure Virtualization. The Security Lab is distributed between two locations: one part of the group sits in the main office in Kista, while the other part are pioneers in our relatively new SICS offices in Lund.

SICS Swedish ICT AB, Security Lab
Ideon Science Park
Building Beta 2 3v
Scheelevägen 17
SE-223 70 Lund