Exercise for Engineering Secure Software Systems

January 7, 2021: Exercise 7

Henning Schnoor

Institut für Informatik, Christian-Albrechts-Universität zu Kiel

Exercise Group Reorganization

interested?

- use `https://wetalk.informatik.uni-kiel.de/home/channels/inf-esss`
- mail me any changes

Recap

Attack on Needham-Schroeder

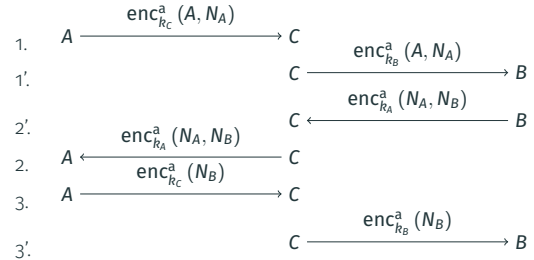
protocol

1. $A \to \cancel{B}\ C \quad \mathrm{enc}^{\mathrm{a}}_{\cancel{k_B}\ k_C}(A, N_A)$
2. $\cancel{B}\ C \to A \quad \mathrm{enc}^{\mathrm{a}}_{k_A}(N_A, \cancel{N_B}\ N_C)$
3. $A \to \cancel{B}\ C \quad \mathrm{enc}^{\mathrm{a}}_{\cancel{k_B}\ k_C}(\cancel{N_B}\ N_C)$

situation

- Alice starts protocol as initiator with C (attacker)
- Bob starts protocol as responder with Alice
- adjust protocol for this situation

attack (Charlie controlled by $\mathcal{A}$)



consequence

- who is attacked?
- Bob "thinks" only Alice knows N_A and N_B
- C knows N_A and N_B
- what about Alice's point of view?
- suggestions to fix protocol?

Decidability

Rusinowitch-Turuani Theorem [RT03]
INSECURE is NP-complete

model
INSECURE: instances given

- theorem only covers fixed number of instances
- instance $\hat{=}$ protocol session

reality
unbounded number of sessions

- many users for single server
- different (or same) users at different servers

number of concurrent TLS sessions?

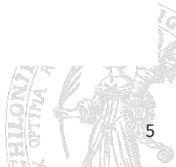
Towards Automatic Analysis

seen in lecture

- formalization of NS protocol must contain sessions to find attack
  - sender instance of $A \to C$
  - receiver instance of $A \to B$
- unsatisfying: this “tells the algorithm where to look”

possible way out: over-approximate

- observation: more instances only make the situation worse (more insecure)
- therefore: let algorithm analyze the following:
  - sender instance of $A \to B$, $A \to C$, $B \to C$
  - receiver instance of $A \to B$, $A \to C$, $B \to C$
- issues?



Discussion: Tasks for this week

Exercise

Task (the FFGG prototocol: too complicated?)

Can you come up with a simpler protocol that is secure when only one session is running, but becomes insecure if the adversary can start as many instances as she wishes? Is there an "advantage" of the ffgg protocol (as an example illustrating the need for the analysis of parallel sessions) over your example?

Exercise

Task (unbounded instances formalization)

Specify the Needham-Schroeder protocol as an instance of the decision problem UNBOUNDED-INSECURE, and show that it is insecure in this formalization. Discuss the differences between expressing the protocol using this formalism compared to the earlier formalization using the decision problem INSECURE.

Note: You do not need to make your constructions formal. The goal of this exercise is to get a good understanding on how a formal definition of INSECURE (which we did not fully state in the lecture) would look like.

Exercise

Task (Rusinowitch-Turuani with specified maximal number of sessions)

We saw in the lecture that the "unbounded session" version of INSECURE is undecidable. A weaker version of that problem can be obtained by allowing instances to INSECURE to be accompanied by a maximal number of copies in which the adversary may start the corresponding protocol instance (we assume a mechanism that automatically renames variables to ensure that they are "local" to the copy in which they are used). Does the "positive" part of the Rusinowitch-Turuani theorem still hold for this generalization?

Hint: You are not expected to give a formal proof of your conjectures, an informal justification suffices. Also, be explicit about how the "maximal number of copies" is specified in the input to your generalized problem.