CryptoVerif - Practical Session 2

June 2023

1 The signed DH protocol

The goal of this session is to complete the *template-signedDH.ocv* file with a full modeling of the signed DH protocol, including secrecy and authentication properties that should automatically prove.

The model will be a "basic" version, we only model two fixed honest participants A and B. A and B can initiate an arbitrary number of sessions, and the attacker can control who A and B will try to talk to.

The template file already contains the main process declaration, starting a process for A with fixed key skA and processB with skB for B.

In addition, the template contains most of the preamble definitions that you should use (types, macro expansion for crypto functions,...).

1. Define the pkI macro process that should fill a table (to define) corresponding to the public key infrastructure: given a hostname, one can retrieve the corresponding public key.

2. Define the processA corresponding to an initiator behavior, with the attacker chosing the hostname of the peer.

3. Define the processB similarly.

4. Add events into processA and B for the authentication queries. add the authentication queries. Autoprove it with Cryptoverif. (refer to the slides if you do not know where to add the events)

5. Whitin processA and processB, store whitin variables keyA and keyB respectivekt the final key when it is a key for an honest session between A and B, and add the corresponding queries checking that all values stored whitin keyA and keyB are secret. Autoprove it with Cryptoverif.

6. Extract from CryptoVerif output the final probabability bound for the secrecy of keyA. Try to roughly give meaning to each part of the sum as well as their corresponding factor, w.r.t. to an "intuitive" security argument of the signed DH protocol.

7. Do a similar thing for the query checking that B is authenticated to A. Do you think the bound is optimal ? If it is not, can you find a way to test it ?