COMP 522: Privacy and Security

Lab session 8

Lecturer: Alexei Lisitsa

Introduction to ProVerif

The goal of this session is to try one of the modern formal verification tools for security protocols, *ProVerif* by Bruno Blanchet et al. The tool with all related documentation can be found at

https://prosecco.gforge.inria.fr/personal/bblanche/proverif/

Please notice that for the purpose of this session we will use Online Demo for *ProVerif* available at http://proverif20.paris.inria.fr/

If you wish to install the tool on your own machines the detailed instructions can be found in the User Manual at
https://prosecco.gforge.inria.fr/personal/bblanche/proverif/manual.pdf

Task 1: In a web browser go to http://proverif20.paris.inria.fr/ (Online Demo of ProVerif), load one of the available protocols and try to verify it. Have a look at the chosen protocol specification and the results of verification. While some of it can be understood without any preliminary knowledge of ProVerif, or reference to the User Manual, many details perhaps would need further explanation and clarification.

Task 2: Go through the Chapter 2, Getting Started (pages 7–9) of the User Manual at https://prosecco.gforge.inria.fr/personal/bblanche/proverif/manual.pdf and execute ProVerif scripts hello.pv and hello_ext.pv discussed there in Online Demo. Actual codes of hello.pv and hello_ext.pv can be found in the documentation package at
https://bblanche.gitlabpages.inria.fr/proverif/proverifdoc2.05.tar.gz

Also, for your convenience I have uploaded it to the web page of COMP522.

Please ensure you understand the meaning of the key words **free**, **process** and **query** and the role they play in the verification process.

Explain what the condition to be checked is expressed by the query - Query event(evCocks) ==> event(evRSA).

Task 3: (long, would likely require working after the session in your own time) Go through the Chapter 3, *Using ProVerif*, follow the proposed steps using Online Demo and handshake protocol specified in Proverif script ex_handshake.pv (available in the documentation package and

at the web page of COMP522). Make sure you understand all parts of the *handshake* protocol specification, in particular, what is **let** contruction which is used there. Make sure you understand how the example *handshake* protocol is verified, and what is an attack found by ProVerif.

Task4: Return to the main page of Online Demo and load Needham-Schoeder-Lowe protocol. This is corrected version of the protocol. Could you edit it to get an original Needham-Shroeder protocol and using ProVerif demonstrate an attack on the original protocol?