Security Web
2020/2021

P.Lafourcade

Verification Tools

SETUP All files needed for this session are avaible on the lecture web page:
`https://sancy.iut-clermont.uca.fr/~lafourcade/SecModels.html`
You also find on this page the link to all the tools, and the parameters to configure your environement for AVISPA.

Recall All binaries are in `/opt` on `turing.local.isima.fr` (172.16.32.21) server:

- `/opt/avispa-1.1`
- `/opt/scyther-linux-v1.1.3`
- `/opt/proverif2.00`

All tools are installed in you path by default with the following commands:

- `avispa`
- `scyther-gui.py`
- `proverif`

For using AVISPA the command line is `avispa`, for scyther is `scyther` and for proverif is `proverif`. With AVISPA you need to specify the tool used for instance by typing
"`avispa toto.txt --output=. --ofmc`"
You should use proverif in Horn clause using the following command
"`proverif -in horn toto.txt`"

Exercise 1 (AVISPA)

1. Use AVISPA (`http://www.avispa-project.org/`) to analyze Needham Schroder protocol, called NSPK with the 4 Tools (OFMC, Cl-Atse, SATMC and TA4SP) using the web interface and also the command line.

   Hint: For the command line, use avispa –help in an xterm, after having exported the environment variable as indicated on the web site of the lecture.

2. Understand the different results given by the Tools.

3. Copy the NSPK protocol that you have found on the AVISPA webpage in a file called NSPK-Lowe and correct by yourself the protocol with Lowe correction, using AVISPA in command line on your computer.

4. Check your corrected version and compare your result with NSPK-fix given on AVISPA web site.

Exercise 2 (SCYTHER)
Now open in Scyther the file protocol1.spdl downloaded on the lecture webpage using:

```
scyther-gui.py protocol1.spdl
```

a) Verify the security claims in the protocol using Scyther (Press F1). You find several attacks. Explain the attacks: why is the property violated in each case?

b) Copy the protocol file to your own directory, and call it `protocol1fixed.spdl`. In the file, make sure you change 'protocol1' to 'protocol1fixed'. Improve the protocol such that the property now holds, and use Scyther to show that your improved protocol indeed meets the requirements.

   Hint: Examine the first message: $\{\mathbf{R}, ni\}pk(R)$ or compare the protocol to the fixed Needham-Schroeder protocol shown in the lecture.

Exercise 3 (SCYTHER)
Open in Scyther the file protocol2.spdl downloaded on the lecture webpage.
This protocol contains a rather large messages and contains many random numbers (nonces) and hash functions. Not all of these elements are necessary to guarantee the correctness of the protocol.

- Suggest five efficiency improvements (in terms of message size or complexity) for the protocol, and motivate your choice. Test each suggested improvement using Scyther. If any of your suggestions fails, explain why.

Exercise 4 (Proverif with Horn Clauses)

1. Check the file needham.horn with proverif (cf my web page).

   Hint use proverif -help to get some help in a terminal:

   ```
   proverif -in horn needham.horn
   ```

2. Understand the attack found

3. Correct it.

Exercise 5 (OFMC, CL-ATSE)
Consider now the following modified version of NSPK protocol, called NSPKXor.

```
A --> B: {Na.A}_Kb
B --> A: {Nb.xor(Na,B)}_Ka
A --> B: {Nb}_Kb
```

Use OFMC and Cl-Atse to check this protocol in presence of XOR. Explain the result given by the tool and propose a correction if there is a flaw. Notice that the XOR is model in avispa by the binary function `xor(x,y)` that means $x \oplus y$.