Avispa Introduction

André Karge

11. November 2014

Inhaltsverzeichnis

1 AVISPA

1.1 Introduction

AVISPA stands for **A**utomated **V**alidation of **I**nternet **S**ecurity **P**rotocolls and **A**pplications and is a program to analyze cryptographic protocols.
Avispa translates protocols written in HLPSL (High-Level Protocol Specification Language) to the IF-language. The proofer of AVISPA are understanding this language and are interpreting it.

In the paper ***Automated Security Protocol Analysis With the AVISPA Tool*** by Luca Viganò the tool is decribed as follows:
The AVISPA Tool is a push-button tool for the Automated Validation of Internet Security-sensitive Protocols and Applications which rises to this challenge in a systematic way by
i) providing a modular and expressive formal language for specifying security protocols nd properties, the High-Level Protocol Language HLPSL, and
ii) integrating different back-ends that implement variety of automatic analysis techniques ranging from protocol falsification (by finding an attack on the input protocol) to abstraction-based verification methods for infinite numbers of sessions.

Additionally in the paper ***The AVISPA Tool for the Automated Validation of Internet Security Protocols and Applications*** by A. Armando and many others it is described as:
AVISPA is a push-button tool for the automated validation of Internet security-sensitive protocols and applications. It provides a modular and expressive formal language for specifying protocols and their security properties, and integrates different back-ends that implement a variety of state-of-the-art automatic analysis techniques. To the best of our knowledge, no other tool exhibits the same level of scope and robustness while enjoying the same performance and scalability.

The following proofer are provided:

- Cl-AtSe (Constraint Logic based Model Checker)
- ofmc (On-the-fly Model-Checker)
- satmc (SAT based Model Checker)
- ta4sp (Tree Automata based Automatic Approximations for the Analysis of Security Protocols)

1.2 Proofer

1.2.1 Cl-AtSe

- bounded number of loops (bounded number of protocol steps in any trace)
- unification of messages modulo XOR + intruder deduction rules over terms with XOR operator
- unification of messages modulo the exponential + intruder deduction rules over terms with exponential
- Baader Schulz unification algorithm

1.2.2 ofmc

- follows the dolev-yao intruder model
- intruder can read or manipulate received messages and send them on
- performs both protocol falsification and bounded session verification
- guessing attacks on weak passwords

1.2.3 satmc

- performs both protocol falsification and bounded session verification like ofmc

1.2.4 ta4sp

- unbounded protocol verification by approximating the intruder knowledge by using regular tree languages and rewriting

2 Installation Guide AVISPA

- extract the AVISPA Package (note that the tool is for i686 enviroments and make sure that you have to install the needed i686 packages that will be shown if they are not installed)

```
tar -xvzf avispa-package-1.1_Linux-i686.tgz
```

- move extracted folder to /opt

```
sudo mv avispa-1.1 /opt/
```

- make sure avispa is executable

```
sudo chown user:user /opt/
```

- export paths for bash execution

```
export AVISPA_PACKAGE=/opt/avispa-1.1
export PATH=$PATH:$AVISPA_PACKAGE
```

3 Informations

3.1 HLPSL

3.1.1 Channels

- dy = dolev-yao = allows the intruder to change or to fake messages
- ota = over the air = disallows the intruder to change or to fake messages (in AVISPA 1.1 not implemented)

3.1.2 Syntax

- to change a variable a ' has to be added to specify changes (e.g. Step' := 2)
- =|> is a conditional execution of a given right hand side (only executed when left hand side satisfied)
- --|> is a unconditional execution of a given right hand side (after rhs execution: lhs execution)
- an appended _K on message m is an encoding of m if K is a key otherwise if K is an agent m is signed by K
- an appended .A on a message is a concatenation (e.g. A.B.C.D)
- protocol_id = variable pointing on a specific security parameter to identify several security parameters

3.1.3 Proofing

- secret(N,n,A,B) declares that N should only be known by A and B
- witness(A,B,n,N) declares that A created N and sent it to B
- request(B,A,n,N) B checks if N is the message A has declared in his witness

4 Test on deactivating TLS

BootPT

protocol session	TLS off between	step	result	attack
BootPT #1	PT->T	1	unsafe	attack on ID_{PM} between PT and T and attack on K^{P}_{PT} between PT and T
BootPT #4	PT->T, T->PT	4+5	unsafe	attack on K between T and PT and attack on ID_{PT} between T and PT

BootPS

protocol session	TLS off between	step	result	attack
BootPS #1	S->PS	1	unsafe	attack on N_{S} between S and PS
BootPS #2	PS->T, T->PS	2+3	unsafe	attack on H_{S} between PS and T
BootPS #4	PT->T	5	safe	
BootPS #5	T->PS	6	safe	
BootPS #6	PS->S	7	safe	

Auth

protocol session	TLS off between	step	result	attack
Auth #1	S->PS	1	unsafe	attack on N_{S} between S and PS
Auth #2	PS->T, T->PS	2+3	unsafe	attack on H_{S} between PS and T and attack on H_{PT} between PS and T
Auth #4	PT->T	5	safe	
Auth #5	T->PS	6	safe	
Auth #6	PS->S	7	safe	

RevokePS

protocol session	TLS off between	step	result	attack
Auth #1	S->PS	1	unsafe	same as in Auth #1
Auth #2	PS->T, T->PS	2+3	unsafe	same as in Auth #2
RevokePS #3	PS->T	4	safe	

RevokePT

protocol session	TLS off between	step	result	attack
RevokePT #1	PT->T	1	unsafe	attack on unencrypted email in Step 2

Rekeying

protocol session	TLS off between	step	result	attack
Rekeying #2	PT->T, T->PT	2+3	unsafe	secrecy break of K in Step 3

5 Links

Avispa Dokumentation	Link
HLPSL Tutorial	Link
Analysis With the AVISPA Tool	Link
HLPSL Tutorial	Link
The High Level Protocol Specification Language	Link
Hello World in HLSPL	Link
The High-Level Protocol Specification Language	Link
Different Crypto Proofer	Link
A Comparative Analysis of Tools for Verification of Security Protocols	Link
The AVISPA Tool for Automated Validation of Internet Security Protocols and Applications	Link
OFMC: A Symbolic Model-Checker for Security Protocols	Link
OFMC: A symbolic model checker for security protocols	Link
Formal Verification of Authenticated AODV Protocol using AVISPA	Link
Automated Security Protocol Analysis With the AVISPA Tool	Link