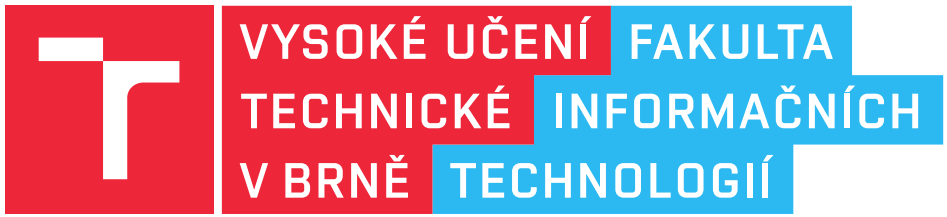
Formal verification and its application in the security of information systems

František Mazura

Brno University of Technology, Faculty of Information Technology
Božetěchova 1/2. 612 66 Brno - Královo Pole



14. december 2017

1. Problem
2. Methods
3. Dolev-Yao Model
4. AVISPA
5. HLPSL

Needham–Schroeder protocol

1. $Alice \rightarrow A, B, R_A \rightarrow Trent$
2. $Trent \rightarrow \{R_A, B, K, \{K, A\}_{K_B}\}_{K_A} \rightarrow Alice$
3. $Alice \rightarrow \{K, A\}_{K_B} \rightarrow Bob$
4. $Bob \rightarrow \{R_B\}_K \rightarrow Alice$
5. $Alice \rightarrow \{R_B - 1\}_K \rightarrow Bob$

Needham–Schroeder protocol

1. $Alice \rightarrow A, B, R_A \rightarrow Trent$
2. $Trent \rightarrow \{R_A, B, K, \{K, A\}_{K_B}\}_{K_A} \rightarrow Alice$
3. $Alice \rightarrow \{K, A\}_{K_B} \rightarrow Bob$
4. $Bob \rightarrow \{R_B\}_K \rightarrow Alice$
5. $Alice \rightarrow \{R_B - 1\}_K \rightarrow Bob$

Attack: reply 3. $Alice \rightarrow \{K, A\}_{K_B} \rightarrow Bob$

Formal model?

$$Alice \rightarrow \{A, B, N\}_{K_B} \rightarrow Bob$$

$$Bob \rightarrow \{A, B, N\}_{K_A} \rightarrow Alice$$

Formal model?

$$Alice \rightarrow \{A, B, \mathbf{N}\}_{K_B} \rightarrow Bob$$

$$Bob \rightarrow \{A, B, \mathbf{N}\}_{K_A} \rightarrow Alice$$

Methods

- Induction
- Autentization logic
- Model checking

- S(M,R)

- S(M,R)
- Dx (decryption under X's secret key)
- Ey (encryption under any user Y's public key)
- iy (append identifier y to the message)
- dy (delete identifier y from the end of the message)
- d (delete identifier at end of message)

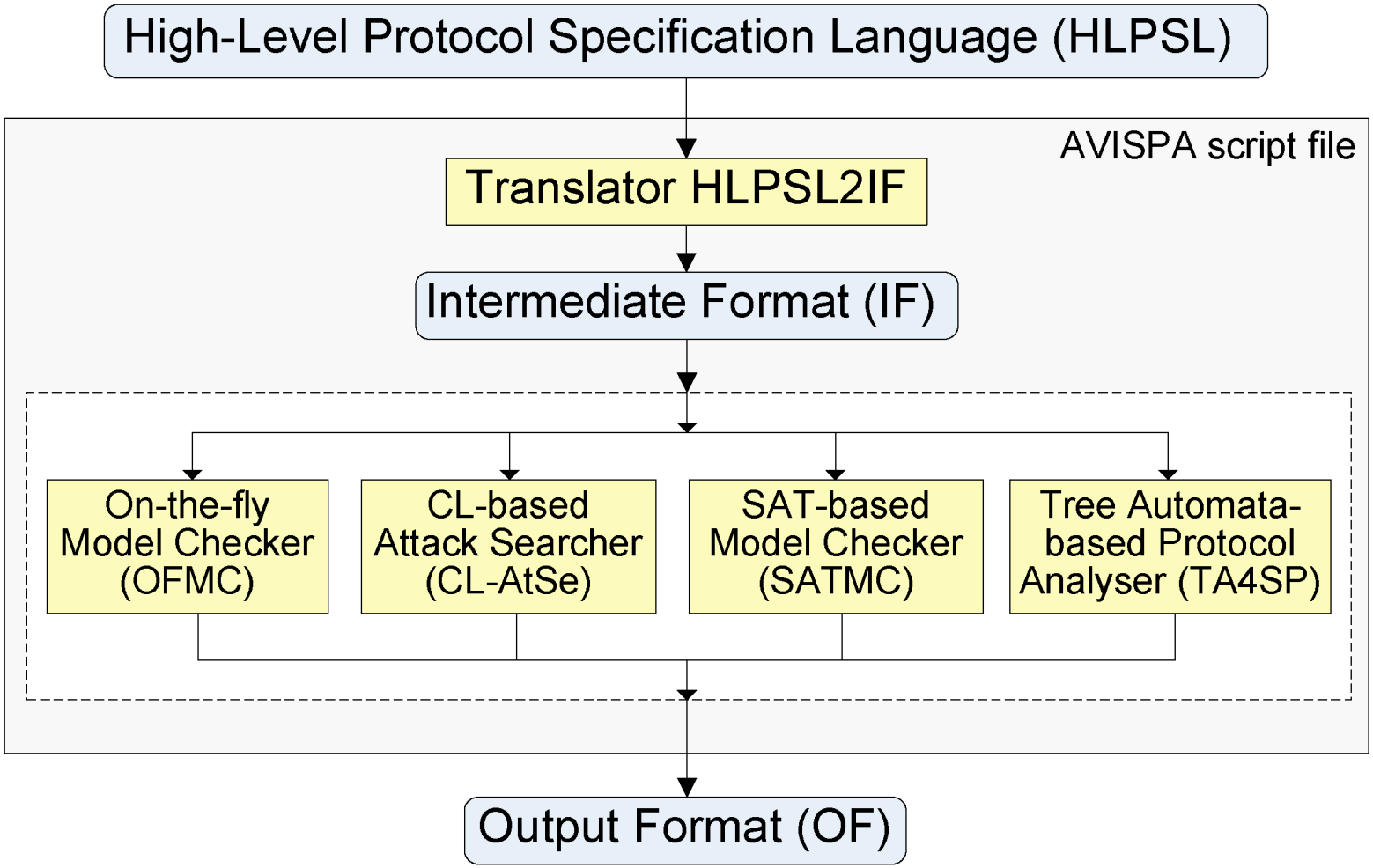
- f(1),f(2),...,f(k)

- f(1),f(2),...,f(k)
- F(i)(M) = f(i)(f(i-1)(...f(2)(f(1)(M))...)

- Dx Ex = ϵ
- Ex Dx = ϵ
- dx ix = ϵ
- d ix = ϵ

- f(1),...,f(r) - two side protocol S $\Leftrightarrow$ *R*

- f(1),...,f(r) - two side protocol S $\Leftrightarrow$ *R*
- gk o ... g2 o g1 o f(1)A,B = id $\Rightarrow$ *insecure*


High-Level Protocol Specification Language (HLPSL)
AVISPA script file
Translator HLPSL2IF
Intermediate Format (IF)
On-the-fly Model Checker (OFMC)
CL-based Attack Searcher (CL-AtSe)
SAT-based Model Checker (SATMC)
Tree Automata-based Protocol Analyser (TA4SP)
Output Format (OF)

```
role alice(
A,B : agent,
K : symmetric_key,
Hash : hash_func,
SND,RCV : channel(dy))
played_by A def=
local
State : nat,
Na,Nb : text,
K1 : message
```

Summary

1. Problem
2. Methods
3. Dolev-Yao Model
4. AVISPA
5. HLPSL