Academic Year: Winter Semester 2021/22

6090: Security of Computer and Embedded Systems

Problem Sheet 10

Formal Analysis of Security Protocols

In this problem sheet, we will

- deepen our knowledge of the Dolev-Yao attacker model and its different uses,
- design and analyse security protocols as well as try to find attacks.

1. Security Protocols and Intruder Model

Exercise 1: ***Intruder Model***

1- In the Dolev-Yao attacker model, an attacker cannot

□ a) eavesdrop all messages

□ b) decrypt all messages

□ c) block all messages

□ d) compose new messages

□ e) create new messages

□ f) decompose messages

2- Assume, in the Dolev-Yao model, the following attacker knowledge:

$$M = \{\{\mathrm{n}_1, \{\mathrm{n}_2\}_{\mathrm{pk}(a)}\}_{\mathrm{inv}(\mathrm{pk}(a))}\}$$

Which messages can the Dolev-Yao attacker generate?

□ a) n_1

□ b) n_2

□ c) $\{\mathrm{n}_1\}_{\mathrm{pk}(a)}$

□ d) $\{\mathrm{n}_2\}_{\mathrm{pk}(a)}$

□ e) $\{\mathrm{n}_1, \{\mathrm{n}_2\}_{\mathrm{pk}(a)}\}_{\mathrm{inv}(\mathrm{pk}(a))}$

□ f) $\{\mathrm{n}_1\}_{\mathrm{inv}(\mathrm{pk}(a))}$

3- Assume, in the Dolev-Yao model, the following attacker knowledge:

$$M = \{\{n_1, \{\mathrm{n}_2\}_{\mathrm{pk}(a)}\}_{\mathrm{inv}(\mathrm{pk}(a))}, \mathrm{inv}\ (\mathrm{pk}(a))\}$$

Which messages can the Dolev-Yao attacker generate?

□ a) $\{\mathrm{n}_1, \{\mathrm{n}_2\}_{\mathrm{pk}(a)}\}_{\mathrm{pk}(a)}$

□ b) $\{\mathrm{n}_1, \mathrm{n}_2\}_{\mathrm{pk}(a)}$

□ c) $\{\{\mathrm{n}_1\}_{\mathrm{pk}(a)}, \{\mathrm{n}_2\}_{\mathrm{pk}(a)}\}_{\mathrm{pk}(a)}$

□ d) $\{\{\mathrm{n}_1\}_{\mathrm{pk}(a)}, \{\mathrm{n}_2\}_{\mathrm{pk}(a)}\}_{\mathrm{inv}(\mathrm{pk}(a))}$

4- Assume, in the Dolev-Yao model, the following attacker knowledge:

$$M = \{\{\mathrm{n}_1, \{\mathrm{n}_2\}_{\mathrm{pk}(b)}\}_{\mathrm{inv}(\mathrm{pk}(b))}\}$$

Which messages can the Dolev-Yao attacker generate?

□ a) $\{\mathrm{n}_1, \{\mathrm{n}_2\}_{\mathrm{pk}(b)}\}_{\mathrm{pk}(b)}$

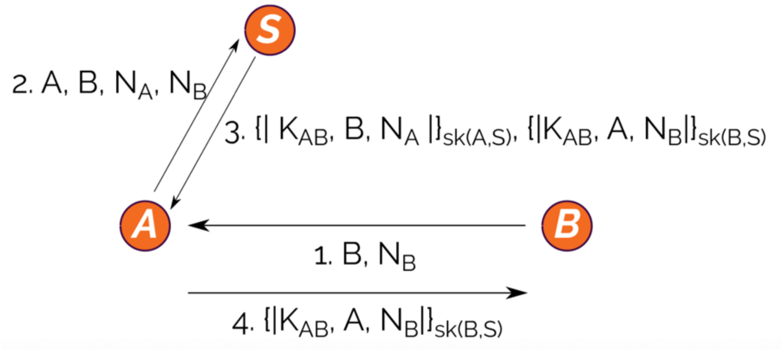
□ b) $\{\mathrm{n}_1, \mathrm{n}_2\}_{\mathrm{pk}(b)}$

□ c) $\{\{\mathrm{n}_1\}_{\mathrm{pk}(b)}, \{\mathrm{n}_2\}_{\mathrm{pk}(b)}\}_{\mathrm{pk}(b)}$

□ d) $\{\{\mathrm{n}_1\}_{\mathrm{pk}(b)}, \{\mathrm{n}_2\}_{\mathrm{pk}(b)}\}_{\mathrm{inv}(\mathrm{pk}(b))}$

Exercise 2: *Diffie-Hellman*

Recall the key-establishment protocols we designed in the last session which is based on an honest key-server S who has a shared key $\mathrm{sk}(A, S)$ with every agent A:


S
2. A, B, N_A, N_B
3. {| K_AB, B, N_A |}_sk(A,S), {|K_AB, A, N_B|}_sk(B,S)
A
B
1. B, N_B
4. {|K_AB, A, N_B|}_sk(B,S)

Also, recall the Diffie-Hellman key-exchange protocol discussed in the lecture:

- A and B agree on a DH group (g, p)
- A generates large x and sends half-key $X = g^x mod\ p$ to B
- B generates large y and sends half-key $Y = g^y mod\ p$ to A
- A and B compute the key $k = Y^x mod\ p = X^y mod\ p$

1. Combine both schemes, using the key-server to authenticate the exchange.
2. Argue why your Diffie-Hellman based protocol offers stronger security than the key-exchange protocols discussed in the lecture. Consider that the intruder is able at some point to compromise the honest key-server and find out all long-term keys $\mathrm{sk}(A, S)$.

2. Analyzing Security Protocols

In the following exercise, you will deepen your knowledge of the impact of making assumptions and using abstractions when analyzing security protocols.

Exercise 3: *(Non-)Attack-Preserving Assumptions*

In protocol analysis, making assumptions or abstracting certain things away can be very helpful. Some assumptions, however, can exclude attacks at analysis time. We call these assumptions non-attack-preserving. Using non-attack-preserving assumptions is a trade-off.

1. What are some arguments for and against the use of non-attack-preserving assumptions or abstractions?
2. In the lecture we have made the assumption that when A and B receive messages, they “know” what protocol they belong to. Do you think this assumption is reasonable? Do you think it is attack-preserving?

3. The Dolev-Yao Attacker Model

In the following exercise, you will deepen your knowledge of the Dolev-Yao attacker model for analysing security protocols.

Exercise 4: *Generating Messages as Dolev-Yao Attacker*

Consider the following three steps of a security protocol. In this protocol, A and B rely on a trusted third party S:

1- $A \rightarrow B: A, \mathrm{N}_A$
2- $B \rightarrow S: B, \mathrm{N}_B, \{|A, \mathrm{N}_A|\}_{K_{BS}}$
3- $S \rightarrow A: \{|B, \mathrm{N}_A, K_S|\}_{K_{AS}}, \{|A, K_S|\}_{K_{BS}}, \mathrm{N}_B$

We assume that A and S share the symmetric key K_{AS} and B and S share the symmetric key K_{BS}. A and B generate the nonces N_A and N_B, respectively.

Consider the following messages (a to c) generated by a Dolev-Yao attacker. For each message, indicate how many of the first three steps of the above described protocols the attacker would need to see before he or she could generate the message.

Write 0 if the attacker could generate the message without seeing any protocol message and ∞ if the Dolev-Yao attacker could never generate the message. Each answer should be one of $\{0,1,2,3,\infty\}$

a) N_B
b) $\{|\ \mathrm{N}_A|\}_{K_{AS}}$
c) $\{|A|\}_{K_X}$ where K_X is a fresh key

References

1. D. Dolev and A. C. Yao. On the Security of Public Key Protocols. Symposium on Foundations of Computer Science, 0:350–357, 1981. The paper is available at: https://www.cs.huji.ac.il/~dolev/pubs/dolev-yao-ieee-01056650.pdf
2. M. Huth and M. D. Ryan. Logic in Computer Science: Modelling and Reasoning About Systems. Cambridge University Press, 2004.