CHRISTIAN-ALBRECHTS-UNIVERSITÄT ZU KIEL Priv.-Doz. Henning Schnoor
Institut für Informatik · AG Software Engineering

Engineering Secure Software Systems Winter 2020/21
Exercise Sheet 4

issued: November 24, 2020 **due:** December 3, 2020

Exercise 4.1, Formal Representation of the Woo Lam Protocol (10 Points)

Study the authentication protocol by Woo and Lam (see slide 17 of the lecture from November 10).

1. Specify the protocol as sequence of receive/send actions, once in the intended execution between Alice and Bob, and once in a form that allows to model the attack introduced in the lecture.
2. Specify the attack on the protocol formally.
3. How can we modify the protocol in order to prevent this attack?

Exercise 4.2, Otway Rees Protocol (10 Points)

Consider the following protocol (Otway-Rees-Protocol):

1. $A \to B$ $[M, A, B, \mathsf{enc}^{\mathsf{s}}_{k_{AS}}([N_a, M, A, B])]$
2. $B \to S$ $[M, A, B, \mathsf{enc}^{\mathsf{s}}_{k_{AS}}([N_a, M, A, B]), \mathsf{enc}^{\mathsf{s}}_{k_{BS}}([N_b, M, A, B])]$
3. $S \to B$ $[M, \mathsf{enc}^{\mathsf{s}}_{k_{AS}}([N_a, k_{AB}]), \mathsf{enc}^{\mathsf{s}}_{k_{BS}}([N_b, k_{AB}])]$
4. $B \to A$ $[M, \mathsf{enc}^{\mathsf{s}}_{k_{AS}}([N_a, k_{AB}])]$
5. $A \to B$ $\mathsf{enc}^{\mathsf{s}}_{k_{AB}}(\mathsf{FAIL})$

1. Why are the subterms M, A, and B in the second message sent both encrypted and as plaintext?
2. Why is the nonce N_b encrypted in message 2?
3. Is the protocol secure? (You do not need to give a formal proof of security or insecurity.)

Exercise 4.3, Security Modeling Issues: Are we Missing Something? (10 Points)

In the lecture, we defined security of a protocol as, essentially, unreachability of a state in which the adversary learns the constant FAIL. However, this FAIL-constant obviously does not have a correspondance in a real implementation of a protocol. In particular, the rules releasing the FAIL-constant are removed from the protocol in a real implementation. As a consequence, a potential security proof of a protocol in our formal model treats a different protocol than the protocol running in a real implementation.

Are there cases where this difference results in an insecure protocol that can be proven secure in our formal model? If this is the case, how can we circumvent this issue?