# Special topics in Logic and Security

Master Year II, Sem. I, 2022-2023

Ioana Leuștean
FMI, UB

# What is a security protocol?

- A security protocol is a set of rules and conventions defining an exchange of messages between two or more agents, with security-relevant goals, as:
  - establishing communication keys
  - agent authentication
  - ensuring secrecy

*Security protocols are three-line programs*
*that people still manage to get wrong.*- Roger Needham

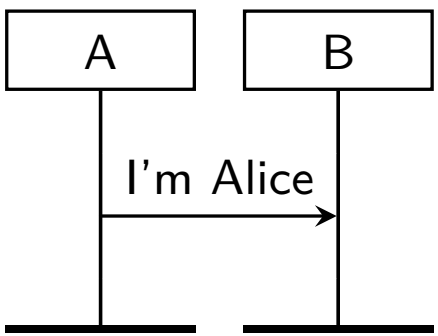
- Even if the cryptography is perfect, a flawed protocol can affect the communication.

# Modelling protocols


## Alice-Bob notation

$A \longrightarrow B$ : "I'm Alice"

## Message sequence charts MSC



# General setting

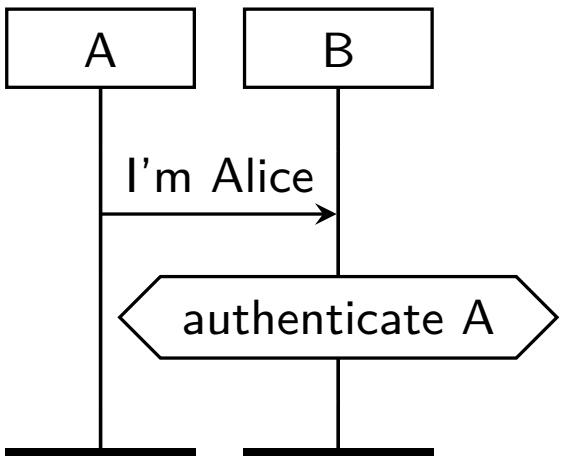
- A security protocol describes few behaviours, which are called *roles* (initiator, responder, server ...). A protocol is complete and unambigous.
- A *session* is a complete execution of a protocol.
- Several protocol sessions may be executed concurrently, an agent can execute any role, possibly in parallel.

# Modelling protocols


## Alice-Bob notation

$A \longrightarrow B$ : "I'm Alice"

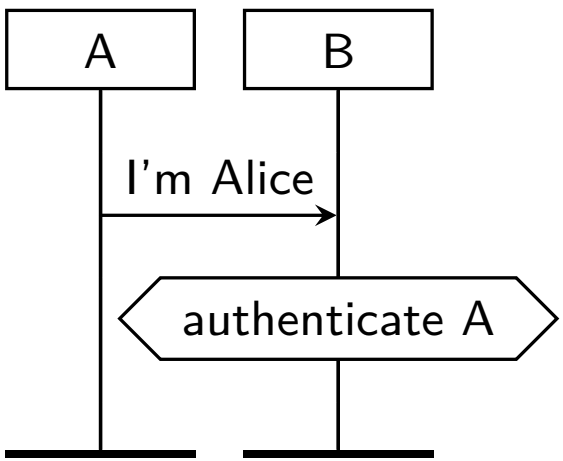
## Message sequence charts MSC




For each role, the vertical axes denotes the order in which the events are executed. The hexagon contains the purpose of the protocol.

# Modelling protocols

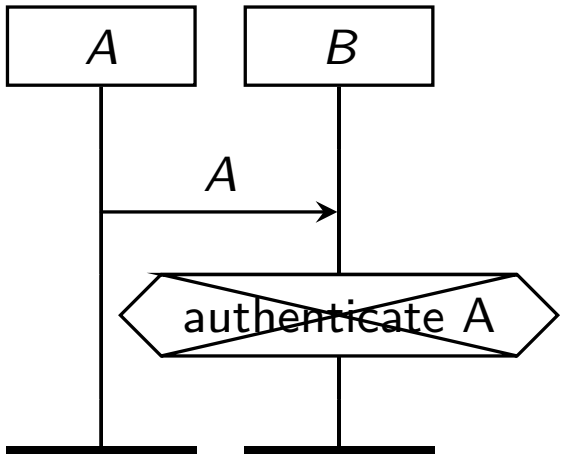
## Message sequence charts MSC



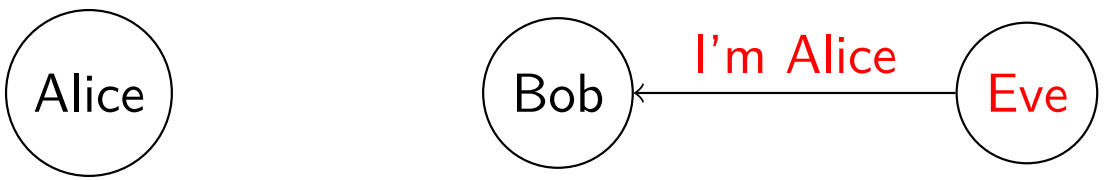

Asssume that *authenticate A* means: *B knows that he speaks with A*.

Is this protocol correct?

# Modelling protocols



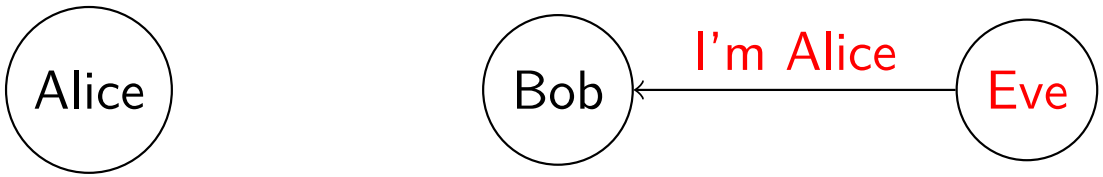

It is possible that Alice is impersonated by Eve!




The intruder Eve impersonates Alice!

$E(A) \longrightarrow B : A$

# Modelling protocols




The intruder Eve impersonates Alice!

$E(A) \longrightarrow B : A$

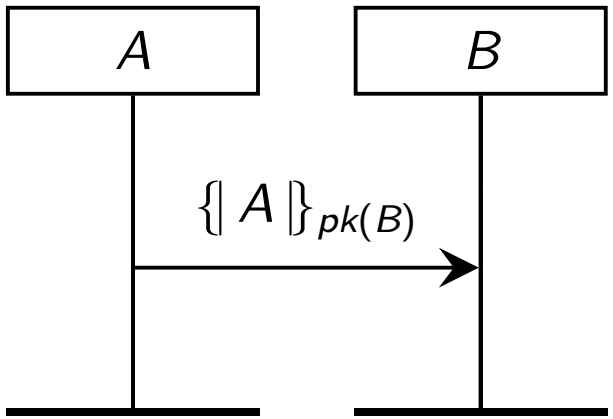
The hostile intruder controls the enviroment where the protocol is executed (the network):

- (s)he can intercept messages,
- (s)he can impersonate any honest agent.

“”””””””””””

# Modelling protocols

$A \longrightarrow B : \{\!|\, A \,|\!\}_{pk(B)}$




- Assume the message is encrypted with $pk(B)$, the public key of $B$.
- The attacker can intercept $\{\!|\, A \,|\!\}_{pk(B)}$ but (s)he cannot decrypt it!

The *black box* (or *perfect cryptography* assumption):
attackers can encrypt or decrypt messages only if they have the right keys.

# The adversary model

The *Dolev-Yao* model:

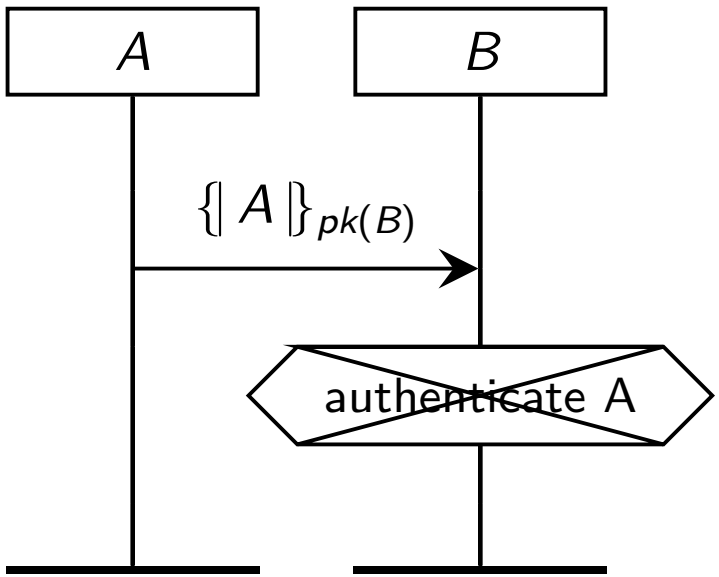
- The adversary has *complete information* over the protocol:
  - s(he) controls the communication channels,
  - s(he) can intercept messages,
  - s(he) has unlimited memory,
  - s(he) can impersonate agents,
  - s(he) can compose messages,
  - s(he) can play the role of an honest agent,

    ...

- but the cryptography *is perfect*:
  the adversary can decrypt a message only if s(he) knows the encryption key.

D. Dolev and A.C. Yao, *On the security of public key protocols*, IEEE Transactions on Information Theory, IT-29 (2): 198–208, 1983.

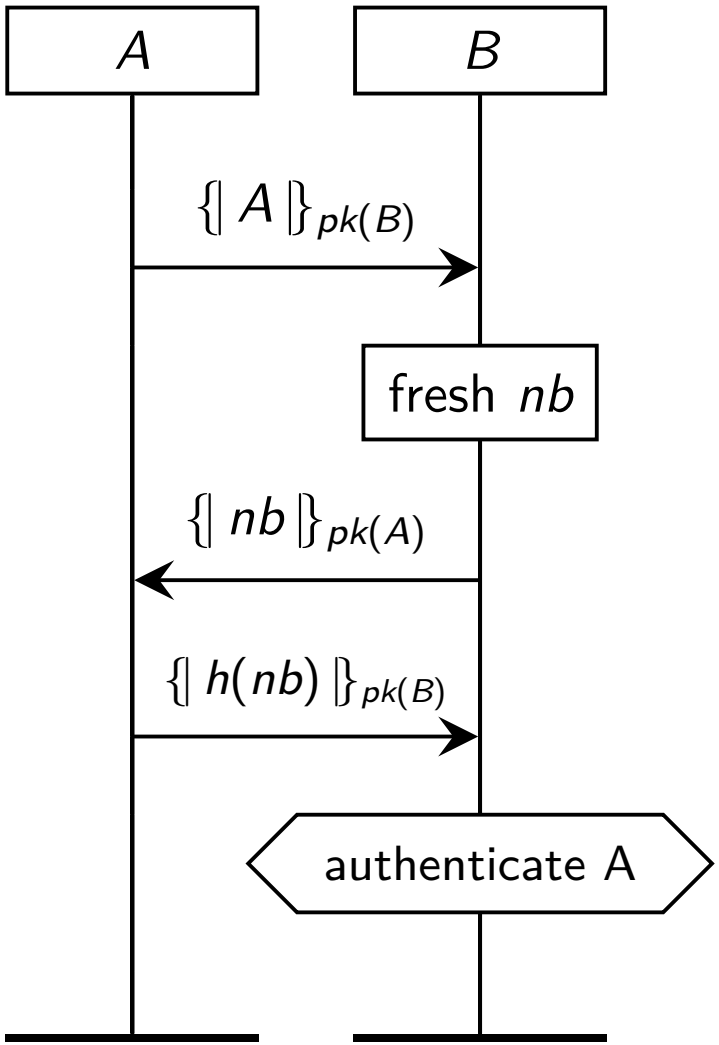
# Modelling protocols




B cannot be sure that he speaks with A!

How do we solve the problem?

# Nonce




- We use a *fresh value*, or a *nonce* (number used once).
- The atatcker can intercept the message $\{\!|\, nb \,|\!\}_{k(A,B)}$ but (s)he cannot extract $nb$ because s(he) does not have the key.
- The attacker cannot send the answer expected by B.

# Example:

Consider the following protocol:

$A \longrightarrow B : A, B, \{\!|\, A, na \,|\!\}_{pk(S)}$

$B \longrightarrow S : \{\!|\, A, na \,|\!\}_{pk(S)}, \{\!|\, B, na \,|\!\}_{pk(S)}$

What is wrong?

$B$ cannot find $na$ since it does not have the private key $sk(S)$, so $B$ cannot compose the second message.

# Example

Consider the following protocol:

$A \longrightarrow B : \{\!|\, A, B, na \,|\!\}_{k(A,B)}$

$B \longrightarrow A : \{\!|\, na, nb \,|\!\}_{k(A,B)}$

where $k(A, B)$ is a symmetric key.

In the following attack $E$ impersonates $B$:

$A \longrightarrow E(B) : \{\!|\, A, B, na \,|\!\}_{k(A,B)}$

$E(B) \longrightarrow A : \{\!|\, A, B, na \,|\!\}_{k(A,B)}$

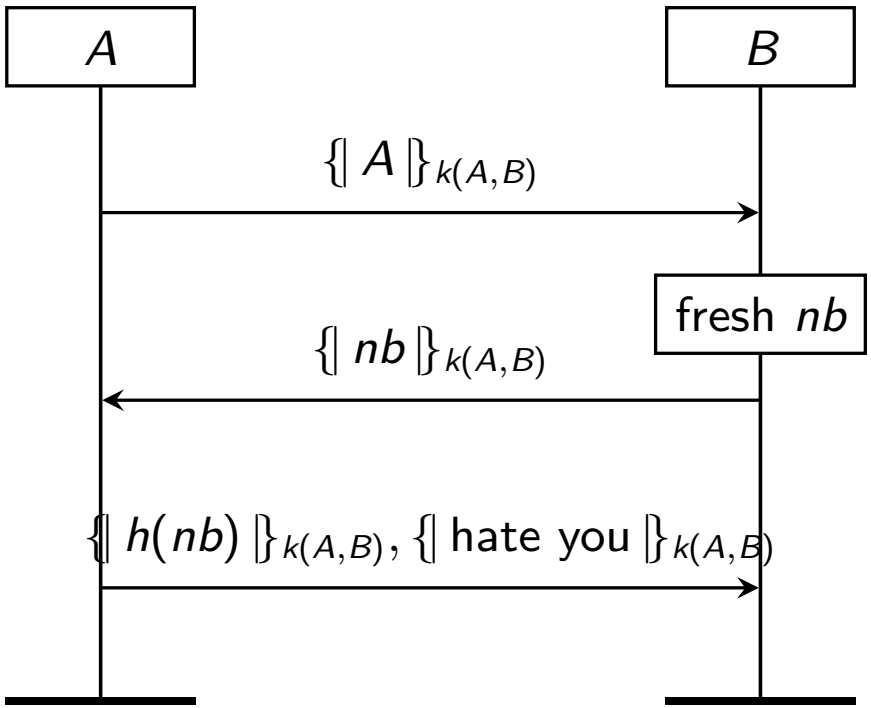
What do you notice?

The attack is *pointless* since $A$ will immediately detect the fact that the received message is composed in a different way. We are not concerned with this kind of attacks!

An *attack* is an exchange of messages between the honest agents and the attacker such that the honest agents *do not detect an anomaly*.

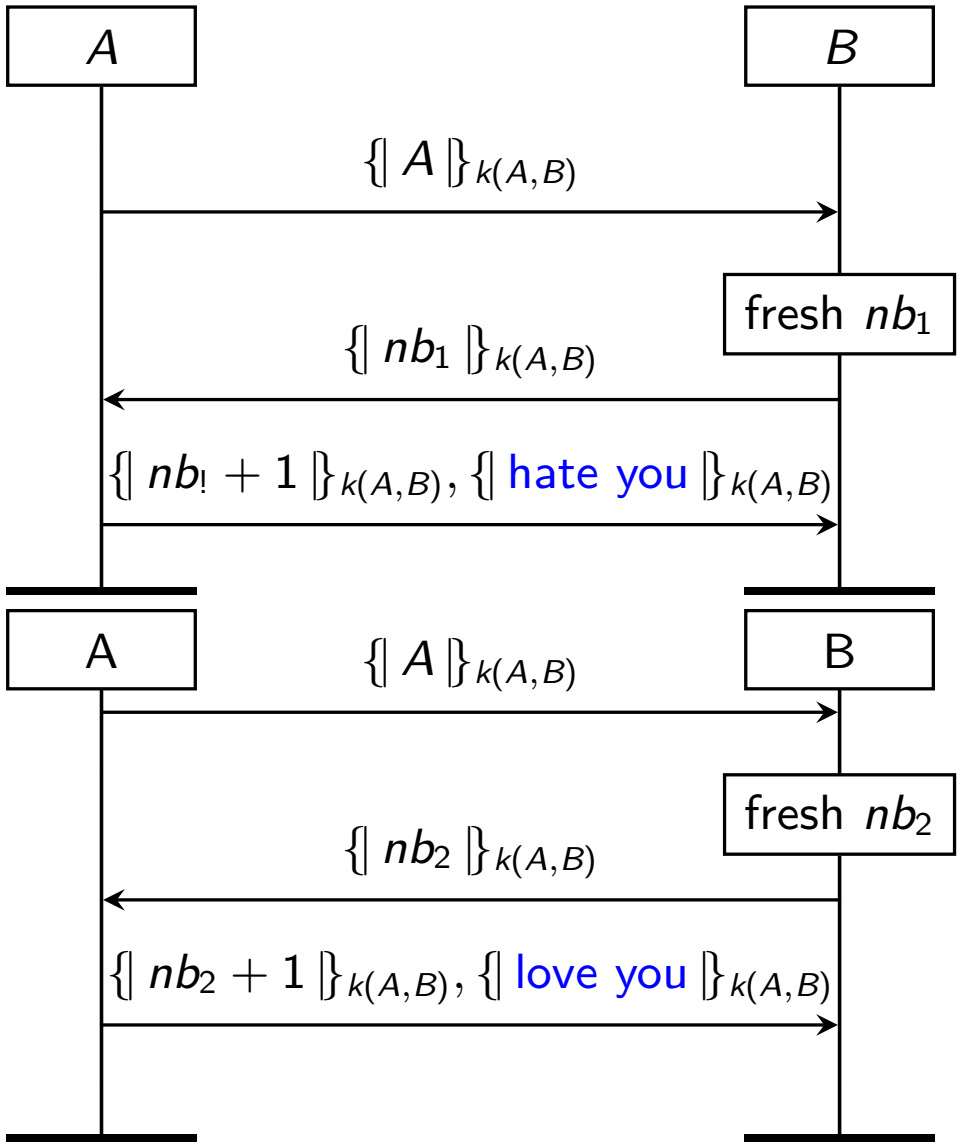
# Modelling protocols

Consider the following protocol:

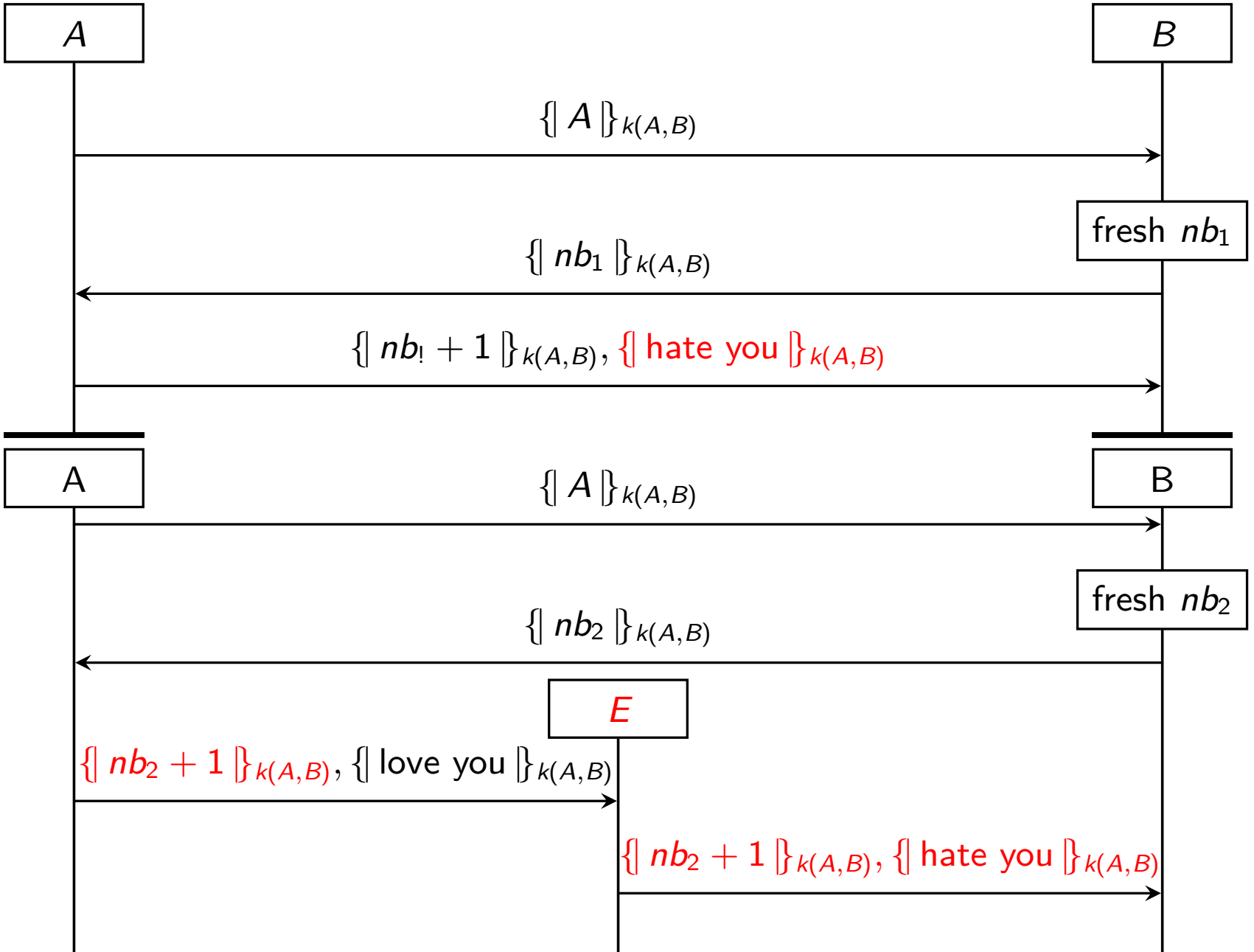



In this session, Bob should reach the conclusion that Alice hates him.

# Modelling protocols: 2 sessions

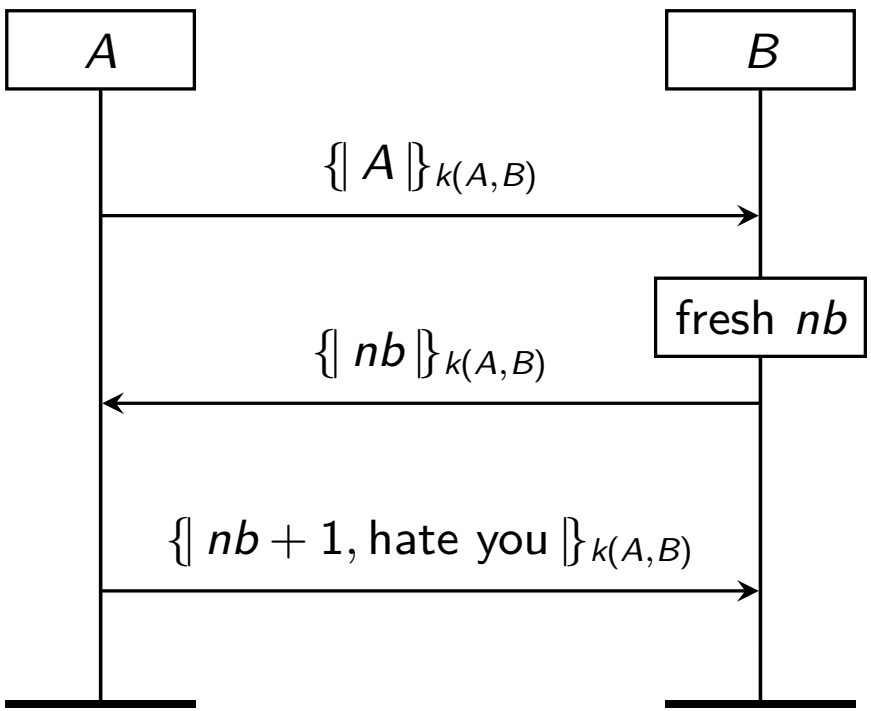


# A replay attack



# Modelling protocols

- În the previous example, the attacker composes a new message from old pieces that (s)he memorized.
- A better version would be:



# The Needham-Schroeder protocol

- In many protocols the symmetric key is a secret established using a public key protocol.

**Needham-Scroeder Public Key Protocol (NSPK)**

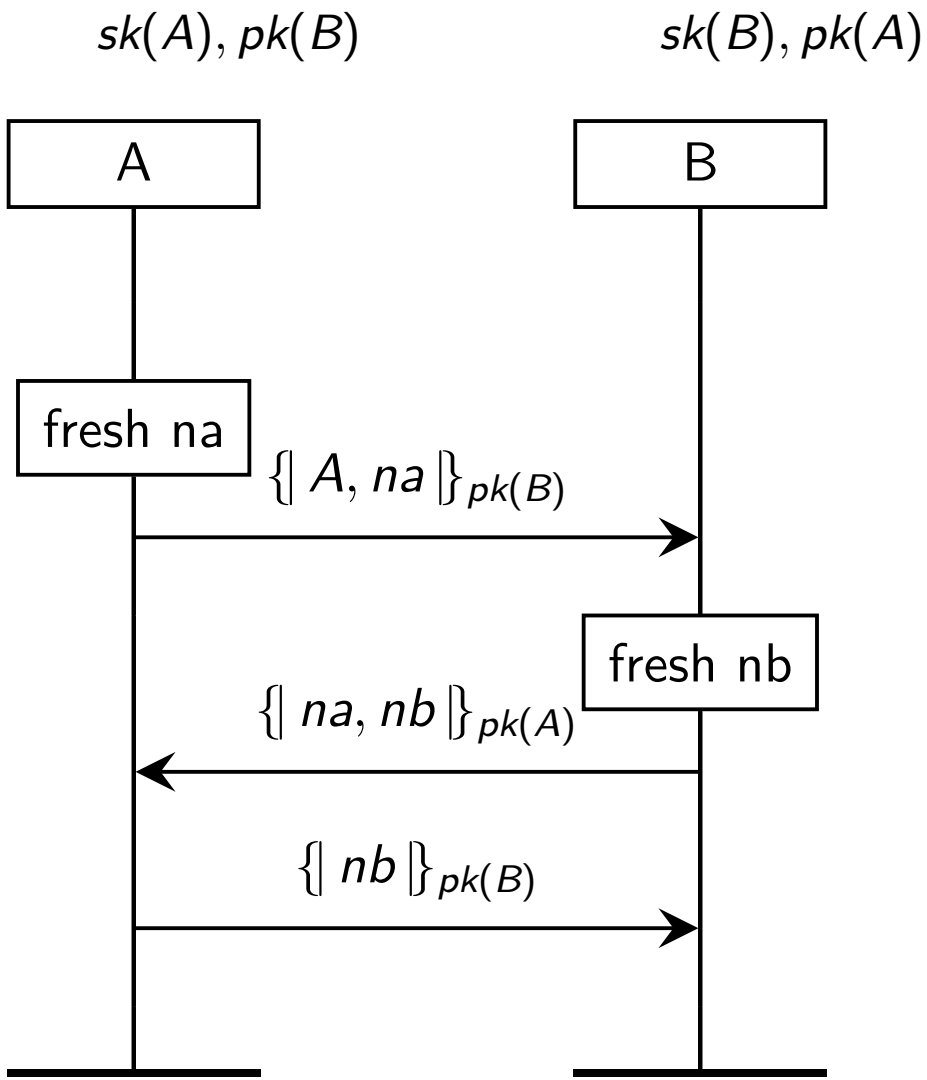
$$\begin{aligned} A &\longrightarrow B: \quad \{\!|\, A, na \,|\!\}_{pk(B)} \\ B &\longrightarrow A: \quad \{\!|\, na, nb \,|\!\}_{pk(A)} \\ A &\longrightarrow B: \quad \{\!|\, nb \,|\!\}_{pk(B)} \end{aligned}$$

The purpose is the mutual authentication of the two participants. After a successful execution:

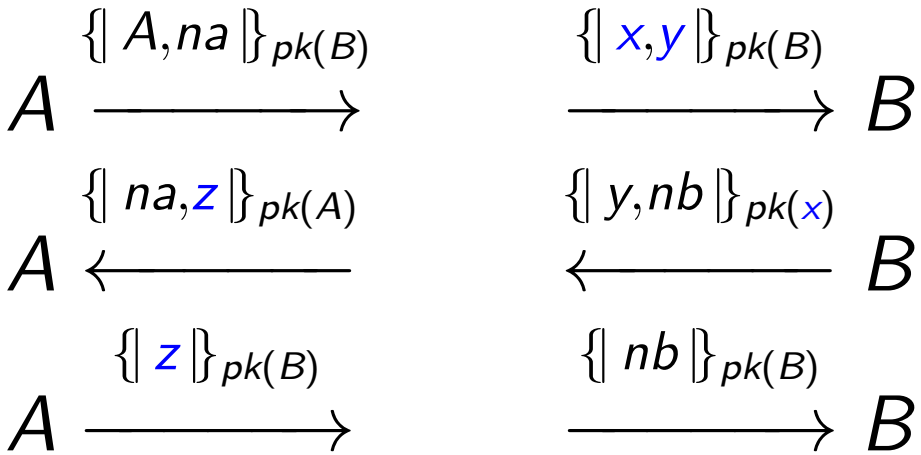
- Alice and Bob should be ensured they have been communicating each other (all the messages have been sent and received by the communication partner),
- *na* and *nb* are known only by Alice and Bob (the protocols guarantees the confidentiality of the nonces *na* and *nb*).

R. Needham and M. Schroeder, *Using Encryption for Authentication in Large Networks of Computers*, Communications of the ACM, 21, pp.393-399, 1978

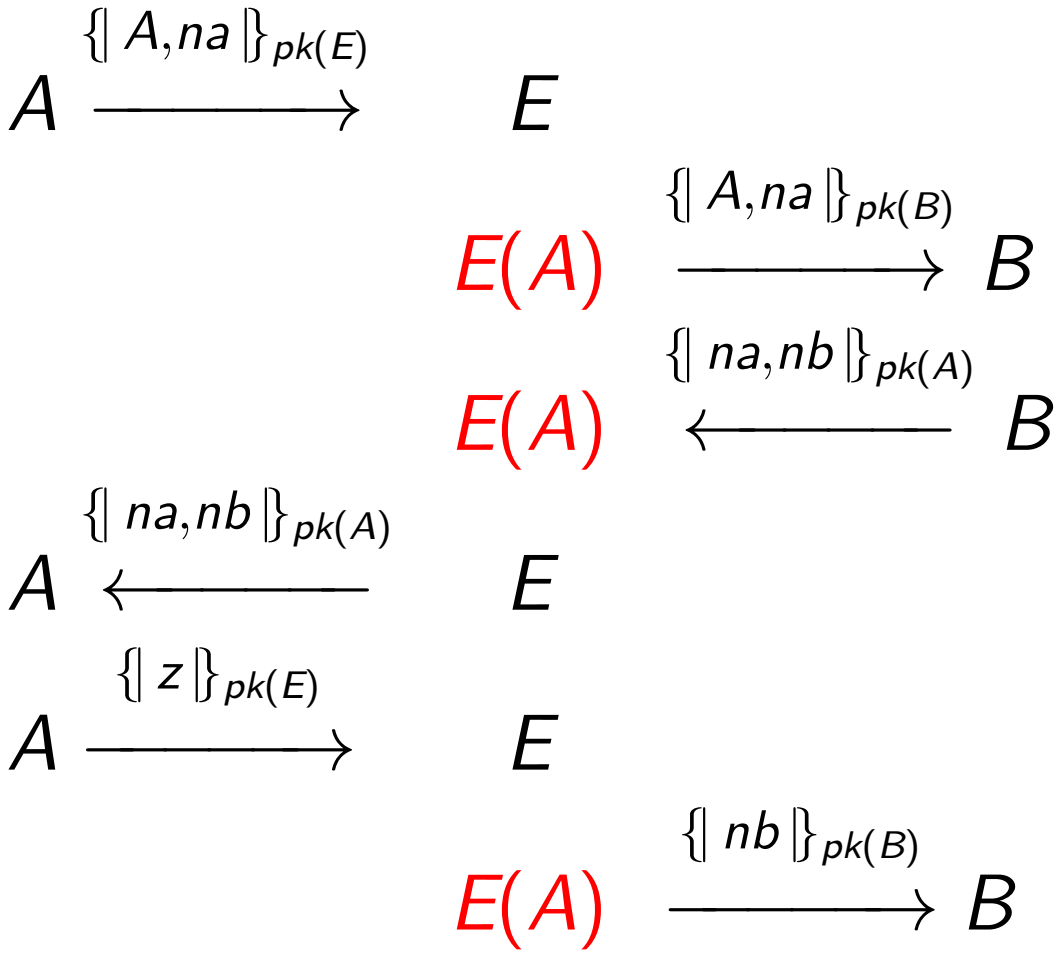
# Needham-Scroeder Public Key Protocol (1978)



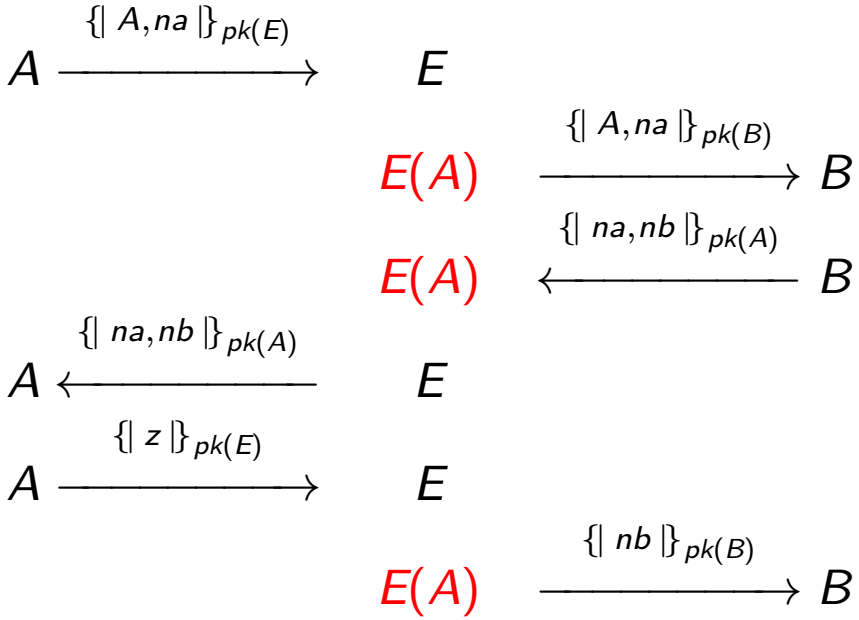
**Needham-Schroeder Public Key Protocol (NSPK)**



**The "man in the middle" attack**



# The "man in the middle" attack on NSPK




- $A$ initiates a session with the (dishonest) $E$,
- $E$ impersonates $A$ when communicating with $B$,
- $B$ belives that he successfully communicated with $A$ but in fact he communicated with $E$. The attack violates authentication from B's point of view, as well as the secrecy of $nb$ ($E$ knows $nb$).

# The Needham-Schroeder-Lowe protocol (1996)

**Fixing NSPK such that the "man in the middle" attack is prevented [Lowe 1996]**

$$\begin{aligned} A \longrightarrow B: &\quad \{\!|\, A, na \,|\!\}_{pk(B)} \\ B \longrightarrow A: &\quad \{\!|\, na, nb, B \,|\!\}_{pk(A)} \\ A \longrightarrow B: &\quad \{\!|\, nb \,|\!\}_{pk(B)} \end{aligned}$$

- $A$ should receive $\{\!|\, na, nb, E \,|\!\}_{pk(A)}$, but $E$ cannot find $nb$.
- $E$ can send $\{\!|\, na, nb, B \,|\!\}_{pk(A)}$ to $A$, but this message does not respect the pattern $A$ waits for.

# Formal analysis of security protocols

- In formal analysis we define and analyze a protocol within a consistent mathematical theory.
- One studies abstract versions of real protocols (for example the (real, computer-network authentication) protocol Kerberos is based on the (academic) protocol Needham-Scroeder.
- Verification based on:
  - logical systems, for example BAN logic,
  - operational semantics,
  - model-checking tools: Proverif, AVISPA, Scyther, Tamarin, ...
  - ...