Case study: Needham–Schroeder protocol

Sara Aschelter, Seweryn Kaniowski

Sapienza Università di Roma

Formal Methods in AI-Based Systems Engineering, Part 2

15 July 2023

Needham-Schroeder protocol [3]

- Application domain: Cryptography.
- Needham-Schroeder Symmetric Key (NSSK) protocol:
  - Symmetric key variant;
  - Shared-key authentication protocol;
  - Based on the existence of a trusted third party;
  - Goal: generate and propagate session keys to establish a secure communication;
  - Authentication using challenge/response technique;
  - The identities of the sender and receiver are communicated inside the messages;

Needham-Schroeder protocol (2)

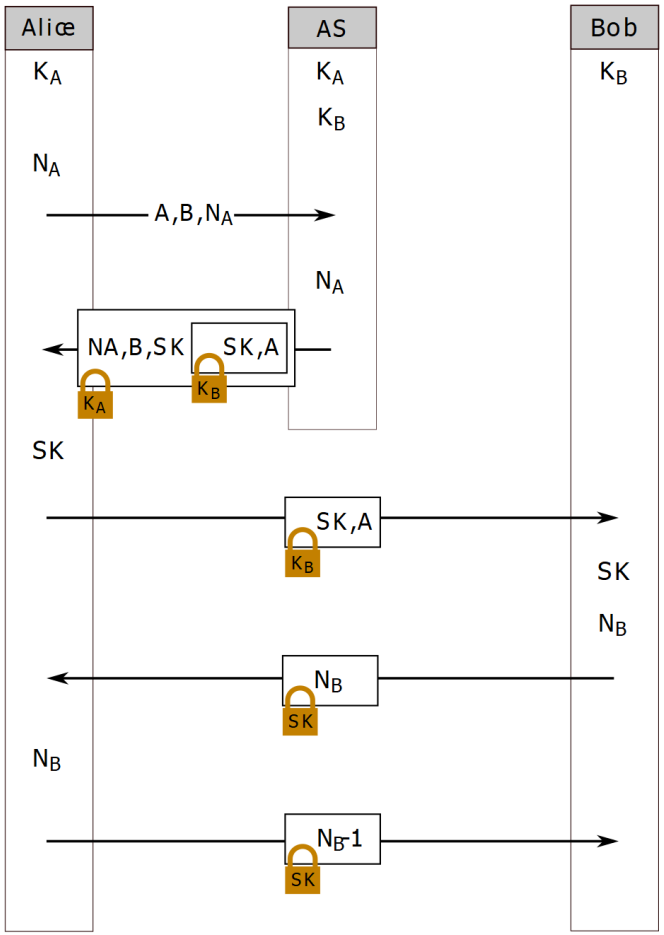


Figure 1: source [2]

- Two parties: Alice and Bob want to communicate, they have their own private keys;
- Third trusted party: Authentication Server (AS), which provides the session keys;
- The protocol is divided into two phases:
  1. Communication between Alice and the AS;
  2. Communication between Alice and Bob;
- Use of nonces: both parties generate fresh random numbers at each execution of the protocol to avoid replies of previously exchanged messages;

Weaknesses of the protocol

After some years from the publication, it was shown that the protocol was vulnerable to:

- Replay Attack [1]:
  1. The attacker listens the communication channel between Alice and Bob and intercepts the exchanged messages;
  2. After a certain amount of time, the attacker can find a session key sent by Alice to Bob;
  3. The attacker knowing the session key and the intercepted messages can resend an old message, containing that session key, as is, without any modifications;
  4. Bob thinks that the message was sent by Alice and replies with a new nonce;
  5. Consequently, the attacker can perfectly reply to the Bob's challenge and being authenticate in place of Alice;

Weakness of the protocol (2)

- Man in The Middle Attack (MITM):
  1. The attacker intercepts the communication between Alice and Bob;
  2. The attacker impersonates Alice to Bob and simultaneously impersonates Bob to Alice;
  3. The attacker can choose to forward the messages to Bob without alteration or manipulate its contents. The same process applies to messages sent by Bob to Alice.
  4. Exchanged all the messages of the protocol the attacker is authenticated in place of Alice;

PRISM model

MDP:

- 3 modules: Alice (sender), Bob (receiver), Cathy (Authentication Server) and 1 player: Eve the attacker;
- States of the model are represented by the values associated to the variables of the protocol (session keys, nonces, etc);
- Built model: States: 7752, Transitions: 17340, Choices: 15896;
- Engine (Hybrid): Nodes: 18166, Time(s): 0.2;

Abstractions

- The values of some variables are not represented explicitly:
  - The variables associated to the keys of Alice and Bob and the variables associated to the nonces can take one of the two values: 1 if they are known and 0 otherwise;
  - The values of the variables associated to the session keys can assume three values: 0 if the session key is unknown, 1 if the session key is fresh, 2 if the session key is old;
- Messages are represented by formulae which are true after the message is sent, and they act as the guards of the commands of receivers;
- We assume that Eve is listening all the time and that intercepts all the messages;

Strategies

In order to show different scenarios of potential attacks we synthesize some strategies starting from a property specification formula.

- 1st scenario: authentication by replay attack using the session key and the 3rd message of the current session.

```
Pmax=? [ F (autE & replay_attack & !knows_kB
          & !knows_kA & skE=1)]
```

Stategies (2)

Strategy		**Bob**			**Eve**					
action	step	stateBe	nBe	skBe	stateE	naE	skE	nE	knows_msg3	initial_round
-	0	0	0	0	0	0	0	0	false	true
[snd_msg1]	1	0	0	0	0	0	0	0	false	true
[rec_msg1]	2	0	0	0	0	0	0	0	false	true
[snd_msg2]	3	0	0	0	0	0	0	0	false	true
[rec_msg2]	4	0	0	0	0	0	0	0	false	true
[snd_msg3]	5	0	0	0	0	0	0	0	true	true
[guess_new_sk]	6	0	0	0	0	0	1	0	true	true
[snd_msg3e]	7	0	0	0	1	0	1	0	true	true
[rec_msg3e]	8	1	0	1	1	0	1	0	true	true
[snd_msg4e]	9	2	1	1	1	0	1	0	true	true
[rec_msg4e]	10	2	1	1	2	0	1	0	true	true
[dec_msg4e]	11	2	1	1	2	0	1	1	true	true
[snd_msg5e]	12	2	1	1	3	0	1	1	true	true
[rec_msg5e]	13	3	1	1	3	0	1	1	true	true

Strategies (3)

- 2nd scenario: authentication by replay attack using a session key of a previous round and the 3rd message corresponding to the same session.

```
Pmax=? [ F (autE & replay_attack & !knows_kB
          & !knows_kA & skE=2)]
```

Strategies (4)

Strategy		**Bob**			**Eve**					
action	step	stateBe	nBe	skBe	stateE	naE	skE	nE	knows_msg3	initial_round
-	0	0	0	0	0	0	0	0	false	true
[snd_msg1]	1	0	0	0	0	0	0	0	false	true
[rec_msg1]	2	0	0	0	0	0	0	0	false	true
[snd_msg2]	3	0	0	0	0	0	0	0	false	true
[rec_msg2]	4	0	0	0	0	0	0	0	false	true
[snd_msg3]	5	0	0	0	0	0	0	0	true	true
[rec_msg3]	6	0	0	0	0	0	0	0	true	true
[snd_msg4]	7	0	0	0	0	0	0	0	true	true
[rec_msg4]	8	0	0	0	0	0	0	0	true	true
[snd_msg5]	9	0	0	0	0	0	0	0	true	true
[rec_msg5]	10	0	0	0	0	0	0	0	true	true
[guess_new_sk]	11	0	0	0	0	0	1	0	true	true
[reset]	12	0	0	0	0	0	2	0	false	false
[snd_msg3e]	13	0	0	0	1	0	2	0	false	false
[rec_msg3e]	14	1	0	2	1	0	2	0	false	false
[snd_msg4e]	15	2	1	2	1	0	2	0	false	false
[rec_msg4e]	16	2	1	2	2	0	2	0	false	false
[dec_msg4e]	17	2	1	2	2	0	2	1	false	false
[snd_msg5e]	18	2	1	2	3	0	2	1	false	false
[rec_msg5e]	19	3	1	2	3	0	2	1	false	false

Stategies (5)

- 3rd scenario: authentication by Man In The Middle Attack.

```
Pmax=?[F(autE & mitm_attack)]
```

Strategy		Alice				Bob				Eve				
action	step	stateA	nA	skA	nbA	stateB	nB	skB	stateE	naE	nbE	skE	nE	mitm
-	0	0	0	0	0	0	0	0	0	0	0	0	0	0
[snd_msg1]	1	1	1	0	0	0	0	0	0	0	0	0	0	0
[rec_msg1]	2	1	1	0	0	0	0	0	0	0	0	0	0	0
[snd_msg2]	3	1	1	0	0	0	0	0	0	0	0	0	0	0
[rec_msg2]	4	2	1	1	0	0	0	0	0	0	0	0	0	0
[snd_msg3]	5	1	1	1	0	0	0	0	0	0	0	0	0	0
[mitm_msg3]	6	1	1	1	0	0	0	0	1	0	0	0	0	1
[rec_msg3]	7	1	1	1	0	1	0	1	1	0	0	0	0	1
[snd_msg4]	8	1	1	1	0	2	1	1	1	0	0	0	0	1
[mitm_msg4]	9	1	1	1	0	2	1	1	2	0	0	0	0	2
[rec_msg4]	10	2	1	1	1	2	1	1	2	0	0	0	0	2
[snd_msg5]	11	3	1	1	1	2	1	1	2	0	0	0	0	2
[mitm_msg5]	12	3	1	1	1	2	1	1	3	0	0	0	0	3
[rec_msg5]	13	3	1	1	1	3	1	1	3	0	0	0	0	3

Model Checking

The aim of the model checking is to analyse the strength of the protocol and the chances that Eve has to be successfully authenticated.

We consider different likelihoods of discovering a key depending on certain factors:

- 0.001 or 0.01, depending on the cryptographic algorithm used (e.g., SHA-256);
- 0.1, if Eve is a highly skilled and resourceful attacker with advanced computational power;
- 0.5, if moderate or significant chance of key discovery;

Model Checking (2)

- Authentication property:

```
P>=1[ F(autA) ]
```

  Evaluates to false, true in the model without Eve.

- Key agreement property:

```
P>=1 [ F (skA>0 & skA=skB) ]
```

  Evaluates to true.

Model Checking (3)

- Key freshness property:

```
P>=1 [ G ((skC=0|skC=2) U skC=1) ]
```

  Evaluates to true.

- Secrecy property:

```
P>=1 [ G (skE=0) ]
```

  Evaluates to false, true in the model without Eve.

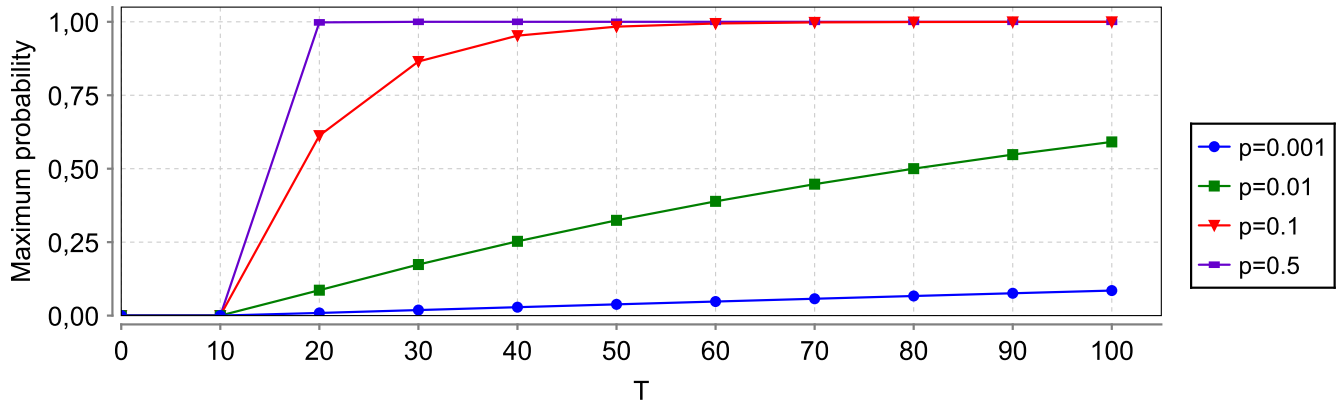
Model Checking (4)

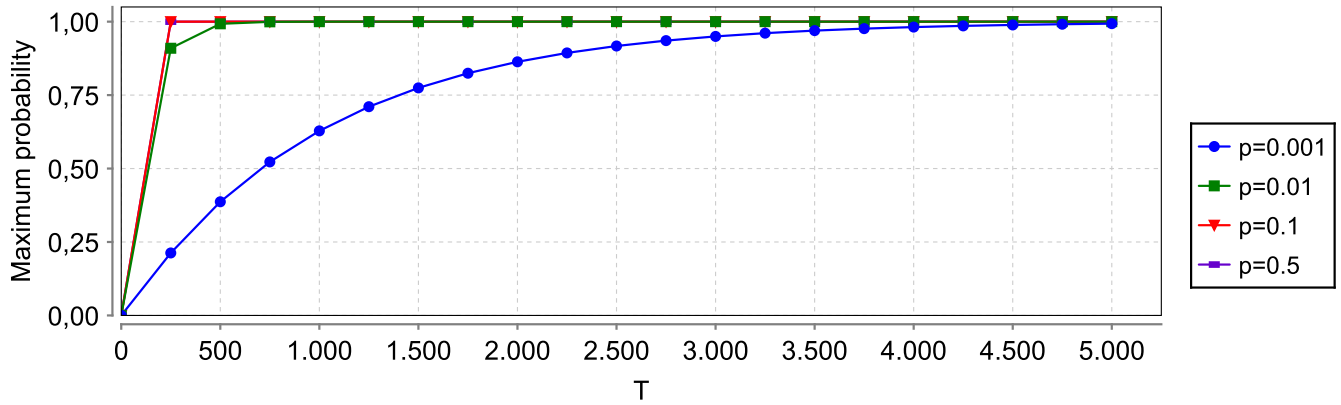
We want to check in the worst case scenario what is the probability that Eve is successfully authenticated up to a given number of steps performing a replay attack:

```
Pmax=? [ F<=T (autE & replay_attack) ]
```

"Maximum probability that Eve authenticates by a replay attack."

Model Checking (5)





References I

[1] Dorothy E. Denning and Giovanni Maria Sacco. "Timestamps in Key Distribution Protocols". In: *Commun. ACM* 24.8 (1981), pp. 533–536. DOI: 10.1145/358722.358740. URL: https://doi.org/10.1145/358722.358740.

[2] Michael F. Schönitzer. *Symmetrical variant of the Needham-Schroeder protocol*. 2007. URL: https://upload.wikimedia.org/wikipedia/commons/e/e7/Symetric_Needham-Schroeder-Protocol_%E2%80%93_linear%2C_schlank.svg.

References II

[3] Roger M. Needham and Michael D. Schroeder. "Using Encryption for Authentication in Large Networks of Computers". In: *Commun. ACM* 21.12 (1978), pp. 993–999. DOI: 10.1145/359657.359659. URL: https://doi.org/10.1145/359657.359659.