Security Protocols and Verification

Design and Analysis of Cryptographic Protocols

Garance Frolla
Ely Marthouret
Ewan Decima

September / November 2025

Contents

1 Assumptions

We assume that at the beginning of the protocol agents A and B know the public key $K_{pub(C)}$ of any agent C. Moreover, we assume that all agents C shared a symmetric key with a trusted server S, named K_{CS}. N_a is a nonce generated by A, and N_b is a nonce generated by B. τ and λ respectively denote a timestamp and a lifetime. A generates K_{AB} with perfect randomness at each session. .

2 Ely the big frog

1. $A \rightarrow B : \{|A, N_A|\}_{K_{AB}}$
2. $A \rightarrow S : \{|B, \tau, \lambda, K_{AB}|\}_{K_{AS}}$
3. $S \rightarrow B : \{|A, \tau, \lambda, K_{AB}|\}_{K_{BS}}$
4. $B \rightarrow A : \{|N_A + 1|\}_{K_{AB}}$

3 Protocol Description

This protocol begins with entity A generating a nonce, denoted as N_A. A then encrypts her identity together with the nonce using the freshly generated session key K_{AB}. The transmitted data is structured as follows: $\{|A, N_A|\}_{K_{AB}}$. This message costs **63**.

After sending the first message, A sends to the honest and trusted server S, using the shared key K_{AS}, the identity of B, a timestamp τ, a lifetime period to confirm the key λ and the session key K_{AB}. The transmitted data is structured as follows: $\{|B, \tau, \lambda, K_{AB}|\}_{K_{AS}}$. This message costs **166**.

Then S, using the shared key K_{BS}, sends to B essentially the same message, but with A replaced by B. The transmitted data is structured as follows: $\{|A, \tau, \lambda, K_{AB}|\}_{K_{BS}}$. This message costs: **166**.

B receives the message $\{|A, \tau, \lambda, K_{AB}|\}_{K_{BS}}$ and obtains the session key K_{AB}. He also learns the validity period λ, starting from time τ, during which A will accept his response. This measure provides protection against ticket theft. Indeed, even if an attacker manages to intercept a ticket, they will not be able to use it after its expiration.

Then B respond to the first message of A, he can decrypt the nonce $\{|N_A|\}_{K_{AB}}$ with the session key. Key confirmation lies in the fact that B sends back $N_A + 1$ to A. In this way, A knows that B has successfully retrieved the key. This allows combining key confirmation with the challenge–response mechanism for the authentication of B with respect to A. The transmitted data is structured as follows: $\{|N_A + 1|\}_{K_{AB}}$. This message costs **12**.

The total cost is: **409**.

4 Security Properties

- *Authentification*: When B receives the first message $\{|A, N_A|\}_{K_{AB}}$, he can't be sure that A sends it to him. But after receiving $\{|A, \tau, \lambda, K_{AB}|\}_{K_{BS}}$ from S he can be sure of the origin of the first message. Indeed, A uses S to authenticate to B. Since the message containing A's identity is encrypted with K_{BS}, B can be sure that A initiated the conversation.

- *Key confirmation*: After receiving the third message, B obtains the session key K_{AB} from S. To confirm possession of this key, B sends back to A the value $N_A + 1$, encrypted under K_{AB}. This proves to A that B has indeed obtained the correct key.

- *Secrecy*: Nobody except for agents A and B know the session key K_{AB}.

- *Integrity*: Each message is encrypted with a symmetric key, ensuring that only an entity possessing the corresponding key (K_{AB}, K_{AS}, or K_{BS}) can modify its contents.