Security Protocols and Verification

Attack of Cryptographic Protocols

Garance Frolla
Ely Marthouret
Ewan Decima

Team: **ASKO OM8464A2**

September / November 2025

Contents

1 Attack on PSS

1. $B \to I(A) : N_B$
2. $I \to A : N_I$
3. $A \to I : \{K_{AI}, A, N_A, N_I\}_{pub(I)}$
4. $I(A) \to B : \{K_{AI}, A, N_A, N_B\}_{pub(B)}$
5. $B \to I(A) : \{|\{N_B, N_A - 1\}_{pub(A)}|\}_{K_{AI}}$
6. $I(A) \to B : \{|N_B - 1|\}_{K_{AI}}$
7. $I \to A : \{|\{N_B, N_A - 1\}_{pub(A)}|\}_{K_{AI}}$
8. $A \to I : \{N_B - 1\}$

2 Attack Description

2.1 Attack Flow

- **Message 1:** B initiates the protocol, believing they are establishing a session with A. B sends their fresh nonce N_B to what they think is A, but the intruder I intercepts this message while impersonating A. The intruder stores N_B for later use in the attack.

- **Message 2:** The intruder I initiates a separate, parallel session with A. I sends their own nonce N_I to A, who believes they are receiving a legitimate protocol initiation from I.

- **Message 3:** A responds by generating a session key K_{AI} (intended for secure communication with I) and their own nonce N_A. A encrypts the tuple $\{K_{AI}, A, N_A, N_I\}$ using I's public key $pub(I)$ and sends it to I. Since I possesses the corresponding private key, they can decrypt this message and obtain K_{AI} and N_A.

- **Message 4:** This is the critical step of the attack. The intruder I impersonates A to B by constructing a fraudulent message. I reuses the session key K_{AI} that A generated for them, but substitutes the nonce N_I with B's original nonce N_B. The message $\{K_{AI}, A, N_A, N_B\}_{pub(B)}$ is encrypted with B's public key, making it appear as a legitimate response from A to B's initial request. B decrypts this message and believes that A has established a shared session key with them.

- **Message 5:** B continues the protocol by computing $N_A - 1$ and encrypting $\{N_B, N_A - 1\}$ with A's public key, then encrypting the result again with what B believes is the shared session key (actually K_{AI}). B sends this double-encrypted message to what they think is A, but I intercepts it.

- **Message 6:** The intruder I, still impersonating A to B, completes the protocol with B by computing $N_B - 1$ and sending it encrypted under K_{AI}. B can verify this response and believes the protocol has completed successfully with A.

- **Message 7:** Meanwhile, I forwards the double-encrypted message from Message 5 to the real A. Since the inner encryption uses A's public key and the outer encryption uses K_{AI} (which A established with I), A can decrypt it successfully.

- **Message 8:** A completes their session with I by computing $N_B - 1$ and sending it encrypted under K_{AI}. Note that A believes they are completing the protocol with I, using nonce N_B that I had sent in Message 2.

2.2 Attack Results

At the conclusion of this attack, both A and B believe they have successfully established a secure session with each other using the session key K_{AI}. However, this session key was actually generated by A for communication with the intruder I. Since I possesses this key, they can decrypt and read all subsequent messages exchanged between A and B.

More precisely:

- A believes they completed the protocol with I and that K_{AI} is shared only with I

- B believes they completed the protocol with A and that K_{AI} (which B thinks is K_{AB}) is shared only with A

- In reality, I knows K_{AI} and can act as a man-in-the-middle, decrypting, reading, and potentially modifying all messages between A and B

This is a classic man-in-the-middle attack that exploits the protocol's failure to adequately bind the session key to both participants' identities in a way that prevents such substitution attacks.