A highly asymmetric key-agreement protocol

Enno Ruijters
August 6, 2015

Overview

The protocol proceeds as follows:

\[ A \rightarrow B : \{k_1\}_{\text{pub}(B)} \]
\[ A \rightarrow B : \{A\}_{\text{pub}(k_1)} \]
\[ B \rightarrow A : \{k_2\}_{\text{pub}(A)} \]
\[ B \rightarrow A : \{k_1\}_{\text{pub}(k_2)} \]
\[ B \rightarrow A : \{B\}_{\text{pub}(k_2)} \]
\[ A \rightarrow B : \{A\}_{\text{pub}(k_2)} \]

Initial knowledge

We assume that A initially knows B’s public key \( \text{pub}(B) \) and that B knows A’s public key \( \text{pub}(A) \).

Data generated during the protocol

\( k_1 \) is a private key generated by A (as well as its associated public key \( \text{pub}(k_1) \)). \( k_2 \) is a private key generated by B (as well as its associated public key \( \text{pub}(k_2) \)).

Protocol description

Alice begins the protocol by generating a new asymmetric keypair \( (k_1, \text{pub}(k_1)) \), encrypting the private key \( k_1 \) to Bob’s public key \( \text{pub}(B) \) and sending it to Bob. She also encrypts her identity to \( \text{pub}(k_1) \) and send this to Bob.

Bob receives and decrypts the private key \( k_1 \) and uses it to decrypt Alice’s identity. He then also generates a new keypair \( (k_2, \text{pub}(k_2)) \), encrypts the private key \( k_2 \) to Alice’s public key and sends it to her. He also encrypts \( k_1 \) and his identity (separately) to \( \text{pub}(k_2) \) and sends these to Alice.

Alice receives the new key \( k_2 \), and uses it to verify that Bob received her \( k_1 \) and sent his own identity. She then encrypts her identity to \( \text{pub}(k_2) \) and sends it to Bob. Bob verifies that this message is correctly encrypted using \( k_2 \).

Security properties

- **Authentication**: The last message received by Bob \( \{A\}_{\text{pub}(k_2)} \) was indeed send by Alice.
- **Confidentiality**: Only Alice and Bob know \( k_2 \), and only Alice and Bob know \( \text{pub}(k_2) \).

Cost:

Every message has a cost of 3, so the total cost is \( 3 \times 6 = 18 \).

Note:

At the end of the protocol, Alice and Bob can use \( k_2 \) directly as an asymmetric key, or they can use it to derive a key for symmetric encryption, e.g. as hash\( (k_2) \).