ESD5 – Fall 2024
Problem Set 5 – Solutions

Department of Electronic Systems
Aalborg University

October 7, 2024

Problem 1

- Too long messages can cause repetition in the cipher which is vulnerable to cryptoanalysis.
- Losing a block means we cannot decrypt the following blocks.

Problem 2

- If one is sent, the authenticity of A can be verified by B, and the integrity of the message, confidentiality as only B can decrypt it. At this point, A can however not verify the authenticity of B, but is sure that no one but B can read or alter the message.
- If both are sent the authenticity of both users can be verified by both. Of course, the properties of confidentiality and integrity remain and are verifiable by both.

Problem 3

- Yes, if the MitM can establish 2 sessions with each user, i.e. create an asymmetric key session with both users then a MitM attack is viable, Fig. 1 shows an example by drawing.

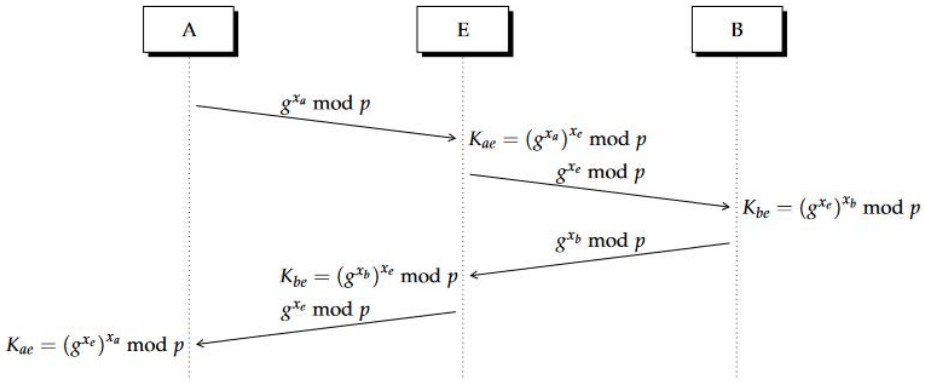

Figure 1: Example of man in the middle attack with Diffie-Hellman. From the paper `https://www.mdpi.com/1099-4300/23/2/226`.

Problem 4

- This is just a proposal to a solution, as long as the authentication is ensured and a new session key can be generated then the principles are correct.

- **Step 1.** A and B send starts a connection.

  Step 2. Authentication via a "challenge"; A sends a random string, and B hashes it with the key and sends it back to A. A can validate B by hashing the challenge and the key together and comparing the hash with what was received. A key point is that the challenge should be random, otherwise, it may be vulnerable to a replay attack. While the initial communication could be encrypted with a key, it is not necessary to authenticate as the one way function property of a hash makes reversing the operation computationally infeasible.

  Step 3. When authenticated, any key generation algorithm could be used e.g. RSA/DH to create the session key.

Problem 5

We use asymmetric encryption to create a secure channel, through which B can define a symmetric key that both can use for a session. By redefining the session key "often" the security may be further enhanced, of course at the cost of more computational complexity.

(a) By adding certificates to the mix, B can be sure that A is who A says he is. This is a critical part of e.g. HTTPS, where websites authenticate their identity through a trusted CA, hence establishing a server-client connection is asymmetricly encrypted. The user can then assume that the website is who they say and that the public key is not false, and data transmissions be done through a symmetrically encrypted channel.