Security

Assignment 3, Monday, September 26, 2011

Handing in your answers: There are two options:

1. Put your solutions in Pim's post box (in the white cabinet, lower door, near the printer at the station-side-end of the corridor on the second floor of the Huygens building). Before handing in, make sure:
   - your names and student numbers are written on the document.
2. Send your solutions by e-mail to `pim@cs.ru.nl` with subject '*assignment 3*'. This e-mail should only contain a single PDF document as attachment. Before handing in, make sure:
   - your names and student numbers are included in the document (since it will be printed),
   - your names are part of the filename (example: PimVullers_FlavioGarcia-3.pdf), and
   - the file is a PDF document.

Deadline: Monday, October 3, 8:30 sharp!

1. **(1 point)** Suppose A and B wish to communicate over an insecure channel. Their goal is to turn it into an *authenticated channel*; this means that the receiver can check:
   - integrity and freshness of data
   - authenticity of the sender.

   Give the internal structure of messages P, Q in:

   $$A \longrightarrow B\colon P \qquad\qquad B \longrightarrow A\colon Q$$

   that are used to achieve this goal. Briefly explain your answer.

2. **(3 points)** The encryption methods you have seen until now were all deterministic. This means: given a message m and a key K, the outcome of the encryption $K\{m\}$ is always the same. In this exercise we look at some of the problems this causes.

   (a) Every Friday evening Alice sends a message to Bob "Will you go out with me tonight?" (in plaintext). Bob, who does not want Eve to be able to read the answer sends an encrypted message back: $K_{AB}\{\textsc{Yes}\}$ or $K_{AB}\{\textsc{Not}\}$, where K_{AB} is a secret key shared between Alice and Bob. Explain why this causes problems for both the confidentiality and the integrity of the message.

   (b) The problem of confidentiality is solved in general by randomizing Bob's message. Instead of simply sending $K_{AB}\{\textsc{Yes}\}$ or $K_{AB}\{\textsc{Not}\}$, Bob chooses a big random number (a nonce) n, and sends $K_{AB}\{\textsc{Yes}, n\}$ or $K_{AB}\{\textsc{Not}, n\}$. Alice ignores n and checks the real answer. Explain why this solves confidentiality, but not integrity.

3. **(3 points)** Which of the following authentication protocols are vulnerable to a reflection attack? If so, write down the attack, using the $A \longrightarrow B : \mathit{message}$ notation. If not, explain why the reflection attack fails. R_A and R_B are random nonces, and K_{AB} is a shared secret key.

   (a)
   1. $A \longrightarrow B \;:\; A, R_A$
   2. $B \longrightarrow A \;:\; R_B, K_{AB}\{R_A\}$
   3. $A \longrightarrow B \;:\; K_{AB}\{R_B\}$

(b)
1. $A \longrightarrow B \;:\; A, R_A$
2. $B \longrightarrow A \;:\; R_B, K_{AB}\{R_A\}$
3. $A \longrightarrow B \;:\; (K_{AB}+1)\{R_B\}$

(c)
1. $A \longrightarrow B \;:\; A, R_A$
2. $B \longrightarrow A \;:\; R_B, K_{AB}\{R_A+1\}$
3. $A \longrightarrow B \;:\; K_{AB}\{R_B\}$

(d)
1. $A \longrightarrow B \;:\; A, K_{AB}\{R_A\}$
2. $B \longrightarrow A \;:\; R_B, K_{AB}\{R_A+1\}$
3. $A \longrightarrow B \;:\; K_{AB}\{R_B\}$

(e)
1. $A \longrightarrow B \;:\; A, K_{AB}\{R_A\}$
2. $B \longrightarrow A \;:\; K_{AB}\{R_B\}, K_{AB}\{R_A+1\}$
3. $A \longrightarrow B \;:\; K_{AB}\{R_B+1\}$

(f)
1. $A \longrightarrow B \;:\; A, K_{AB}\{R_A\}$
2. $B \longrightarrow A \;:\; K_{AB}\{R_B, R_A+1\}$
3. $A \longrightarrow B \;:\; K_{AB}\{R_B+1\}$

4. **(3 points)** Imagine a system where users want to communicate securely using symmetric key encryption. The system has to be be dynamic in the sense that new users should be able to join the system and be able to setup a secure channel with existing users in the system. A protocol is introduced to circumvent the problem that every user needs to setup somehow a new shared key with the latest added users. A trusted server holds a table with (user,key)-pairs where for every user in the system the server has a shared symmetric key. A new record is added when a user joins the system. Now, when two users, let say Alice and Bob, want to communicate, they initially do not share a key with each other. But both share a key with the server (K_{AS} for Alice-Server and K_{BS} for Bob-Server). Alice chooses a key K_{AB} for communication with Bob and sends this encrypted with K_{AS} to the server. The server decrypts the message and encrypts it again with the key K_{BS} (shared key with Bob) and sends it to Bob. Both Alice and the Server add a timestamp T to their message. The protocol is:

1. $A \longrightarrow S \;:\; A, B, K_{AS}\{T_A, K_{AB}\}$
2. $S \longrightarrow B \;:\; K_{BS}\{T_S, K_{AB}, A\}$
3. $B \longrightarrow A \;:\; K_{AB}\{T_B, m\}$

   (a) In the protocol above A and B believe that they communicate with each other. Find an attack on this protocol, and write it down using the $A \longrightarrow B : message$ notation.

   (b) Repair the protocol by doing *only one* change in one message.

   (c) Eventually this protocol seems to work, but when you want to implement it you might face some practical problems. Which drawback does this protocol have from a practical perspective?