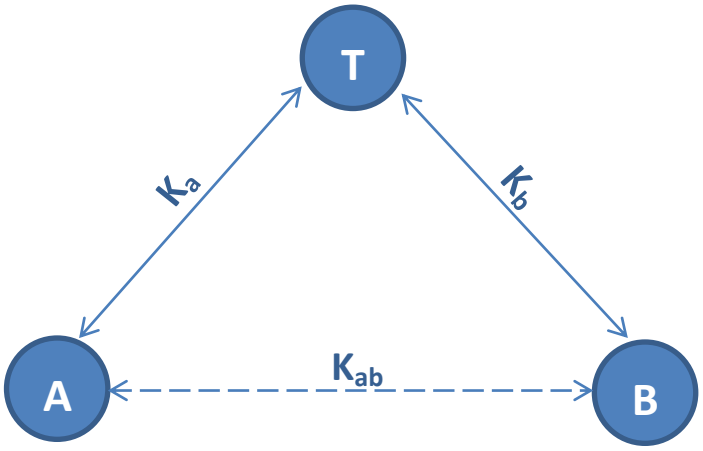
Project SCNS A.A 2014/2015

In this project we have implemented a simple protocol to let users talk confidentially with a server B using a session key each time created by a trusted third party T.

We assumed that the participant to the protocol share long term keys with the trusted inter party, exchanged personally with it in a previous moment. The user and the server store the long term key into a local file called “Key<name>” where <name> is the user identificator. The trusted third party stores the long term keys into files called “Key<name>”; each time it needs to talk with a user, the party accesses the “Database” to retrieve the name of the file in which the key of that user is stored.

The handshake protocol implemented is the following



M1 A->T: **A, B**

M2 T->A: **E_{kb}(A, B, K_{ab}, t) E_{ka}(B, K_{ab}, t)**

M3 A->B: **E_{kb}(A, B, K_{ab}, t)**

In which Kab is the session key that T creates to let a user A and the server B talk to each other and t is the timestamp. We assumed that the protocol works in a system in which clocks are well synchronized and so we assumed that the maximum difference between local and received timestamp must be 2 minutes. If the delay is greater than this quantity, A or B rejects the message as it may not be fresh.

After the session key establishment, A and B can start talking in a confidential way. B offers a service of file storage. The client chooses a file to upload to the server and encrypts its content before sending it. The server receives the encrypted content, decrypts and stores it into a file called “<user>_<date>_<hour>.txt” into the folder “<user>”.

Chiara Caiazza

Martina Troscia