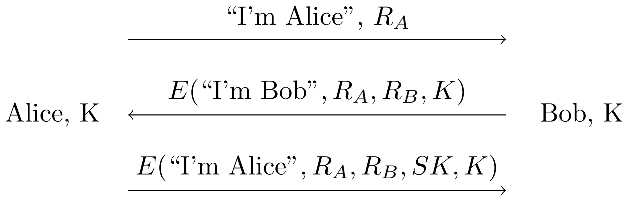
CS 492
Section 70
Spring 2020
Christopher K. Schmitt Assignment 4

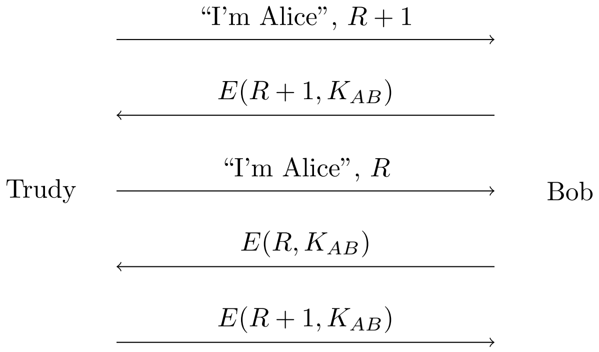
Problem 1

Design a secure mutual authentication protocol based on a shared symmetric key. We also want to establish a session key, and we want perfect forward secrecy. Solve for a protocol that can establish this in 2 to 3 messages


"I'm Alice", R_A
Alice, K
$E(\text{"I'm Bob"}, R_A, R_B, K)$
Bob, K
$E(\text{"I'm Alice"}, R_A, R_B, SK, K)$

Problem 2

Draw the sequence of an attack Trudy can use to convince Bob that she is Alice.



Problem 3

Does Alice authenticate Bob? Justify your answer.

Alice is able to authenticate Bob because in order to compute K, we must have S. S is encrypted by Alice using Bob's public key so only Bob is able to decrypt S and compute K.

Does Bob authenticate Alice? Justify your answer.

No, Bob is unable to authenticate Alice. There is no operation that Alice is required to perform that only she can compute. Trudy has all the information she needs to imitate Alice since anyone can encrypt S for Bob. If CLNT is stored securely however, Trudy shouldn't be able to hijack anything specific to Alice maintained by Bob.

Problem 4

Suppose that nonces RA and RB are removed from the protocol and K=h(S). What effect if any does that have on the security of the protocol?

Removing the nonces in this protocol leaves the communication vulnerable to replay attacks. Without the nonces, the first three messages are identical between instances.

Suppose we change message 4 to HMAC(msgs, SRVR, K). What effect, if any, does this have on the security of the authentication protocol?

Replacing the hash with the HMAC would let Alice know if some third party has tampered with the messages, since computing the HMAC requires knowing the key.

Suppose that we change message three to $\{S\}_{Bob}$, h(msgs, CLNT, K) What effect, if any, does this have on the security of the authentication protocol?

This should have no effect on the security of the protocol. A hash is a one way function, so no information can be extracted from it. Bob can still verify it by computing the hash himself and comparing it.

Problem 5

Why can Alice not remain anonymous when requesting a TGT from the KDC?

Alice must be authenticated, and her identity is contained in the TGT, so she can't remain anonymous.

Why can Alice remain anonymous in the sense of not needing to use her private key when requesting a ticket to Bob (what does she use instead and why is this sufficient)?

Alice has her TGT, which contains her identity behind a key known only to the KCD.

Why can Alice remain anonymous (not needing her private key) when she sends the "ticket to Bob" to Bob?

The ticket to bob is issued by the KCD, which used Alice's TGT to authenticate her. If Bob trusts the KCD, he can trust the ticket to bob without needing to authenticate the person on the other end because the KCD has already done that.