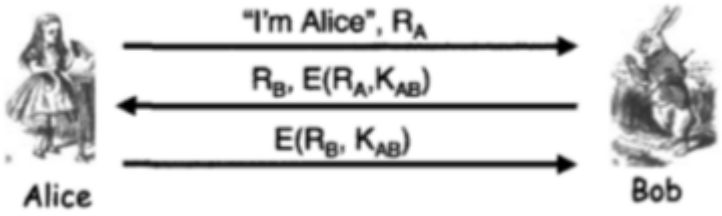
Paul Diaz
HW8

1. Problem 17
   Mutual Authentication protocol based on a shared symmetric key K_{AB}

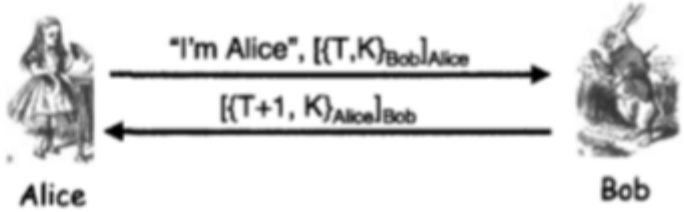


The problem with this scenario is that mutual authentication here is executed twice. Also, it is subject to man-in-the-middle attack. Trudy can start a conversation with Bob by claiming to be Alice and sends a challenge R_A to Bob. Bob then encrypts the challenge R_A and sends it, along with his challenge R_B to Trudy. Trudy actually is not able to continue to reply to Bob, because she doesn't know the key R_{AB}. Trudy then opens another connection to Bob claiming to be someone else, and sends Bob's own challenge R_B, along with $E(R_A, R_{AB})$. Trudy can now go back to the first connection since, she just found out R_{AB}. A modified protocol is provided below to prevent Trudy from attacking.

Alice ————————— "I'm Alice", R_A ————————> Bob
Alice <————————R_A, E("Bob", R_A, K_{AB}) ———————— Bob
Alice ————————E("Alice", R_B, K_{AB})————————> Bob

Here, we encrypt the user's identity together with the nonce, which would be sufficient to prevent Trudy's previous attack since she cannot use a response from Bob for the third message.

2. Problem 18
   Mutual Authentication and key establishment protocol, which employs a Timestamp T and public key cryptography

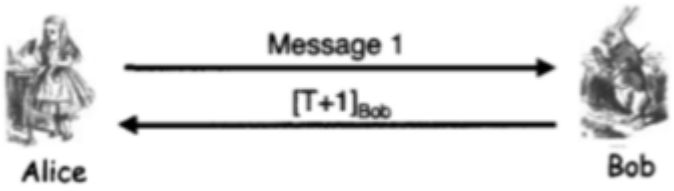


In this situation, Trudy is can recover $\{T, K\}_{Bob}$ by applying Alice's public key. Trudy can then open a connection to Bob and send $\{T, K\}_{Bob}$ in the first message, which in return, Bob will send the key K to Trudy in a form that Trudy can decrypt. Session key K that Alice and Bob must be protected. To prevent such an attack by Trudy, we do this instead.

Alice ————-————"I'm Alice", $[\{T, K\}_{Bob}]_{Alice}$ ————————————> Bob
Alice <————————————$[T+1]_{Bob}$ ———————————— Bob

There's no reason to return the Key K in the second message, since Alice already knows K and the only only purpose of this message is to authenticate Bob. The timestamp in the second message is sufficient to authenticate Bob.

3. Problem 19
   Mutual Authentication and key establishment protocol, which employs a Timestamp T and public key cryptography



a. Message 1: $\{[T, K]_{Alice}\}_{Bob}$
   This protocol is not effective because Alice doesn't tell Bob who she is. Also, Bob cannot verify the signature without knowing who's public key to use to unmasked the message.

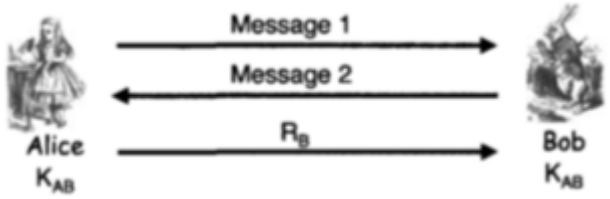
b. Message 1: {"Alice", $[T, K]_{Alice}\}_{Bob}$
   Unlike scenario a, here, Alice tells Bob that she is Alice in the first message — so Bob, can in fact, able to verify who's public key to use and confirm the signature. This protocol has mutual authentication and session key K secure.

c. Message 1: "Alice", $\{[T, K]_{Alice}\}_{Bob}$
   Just like in scenario b, in the first message, Alice tells Bob that she is Alice: only here, not encrypted. This scenario is very similar to scenario b, therefore, we have both mutual authentication, and session key K secure.

d. Message 1: T, “Alice”, $\{[K]_{Alice}\}_{Bob}$
This protocol is subject to replay attack since Alice is not being authenticated. Bob cannot verify the signature, since Bob does not know whose public key to use.

e. Message 1: “Alice”, $\{[T]_{Alice}\}_{Bob}$ and let K = h(T)
Mutual authentication seems to be secure in this protocol. Trudy, however, can do Brute force search to try and guess the time T.

4. Problem 20
Three-message mutual authentication and key establishment protocol, which is based on a shared symmetric key K_{AB}



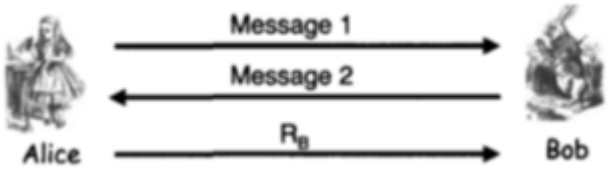
a. Message 1: “Alice”, $E(K, R_A, K_{AB})$ Message 2: R_A, $E(R_B, K_{AB})$
This protocol is not practical. Although Alice tells Bob that she is Alice, Bob cannot verify it because he doesn’t know which key to use.

b. Message 1: “Alice”, $E(K, R_A, K_{AB})$ Message 2: R_A, $E(R_B, K)$
This protocol seems to be okay. Alice tells Bob that she is Alice in the first message. Both get authentication securely in the second and third messages. Both have used session key K secure as well.

c. Message 1: “Alice”, $E(K, R_A, K_{AB})$ Message 2: R_A, $E(R_B, K_{AB})$
Same explanation as in problem b. Alice tells Bob that she is Alice and gets authenticated in the first message, then Bob gets authenticated in the second message. Also session K is secure.

d. Message 1: “Alice”, R_A Message 2: $E(K, R_A, R_B, K_{AB})$
In the first message, Alice tells Bob that she is Alice and sends challenge. Second message is where Bob gets authenticated and third is where Alice gets authenticated. Session key K seems to be secured.

5. Problem 21
   Three-message mutual authentication and key establishment protocol, which is based on public key cryptography



   a. Message 1: {"Alice", K, R_A}$_{Bob}$, Message 2: R_A, {R_B}$_{Alice}$
      In this protocol, Alice tells Bob that she is Alice, but Bob cannot authenticate Alice. Bob, on the other hand, is the only who gets authenticated.

   b. Message 1: "Alice", {K, R_A}$_{Bob}$ Message 2: R_A, {R_B}$_{Alice}$
      I**n the first message, Alice tells Bob that she is Alice and sends her encrypted challenge with session key K. In the second message, Bob tries to authenticate Alice, and Alice authenticates Bob in the third message. Therefore we have secure mutual authentication and session key.**

   c. Message 1: "Alice", {K}$_{Bob}$, [R_A]$_{Alice}$ Message 2: R_A, [R_B]$_{Bob}$
      In the first message, Alice tells Bob that she is Alice but in the second message, Alice is not authenticated. Bob then sends the second message, but Alice has no way to authenticate Bob. Session key is not secure because anyone can use Bob's public key to decrypt it.

   d. Message 1: R_A, {"Alice", K}$_{Bob}$ Message 2: R_A, {R_B}$_{Alice}$
      In the second message, Alice is authenticated by Bob. Alice encrypts the message and gets authenticated in the third message. Just like in protocol c, session key is not secure because anyone can use Bob's public key to decrypt it.

   e. Message 1: {"Alice", K, R_A, R_B}$_{Bob}$ Message 2: R_A, {R_B}$_{Alice}$
      In the second or third message, Alice is not authenticated. Alice sending R_B seems a little weird. I don't know where did she get this challenge.

6. Suppose we replace the third message of the protocol in Figure 9.22 with

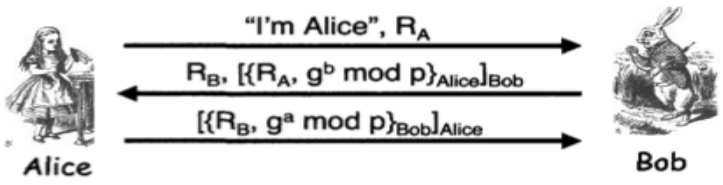


Figure 9.22: Mutual Authentication, Session Key and PFS

$\{R_B\}_{Bob}$, $g^a \bmod p$

a How can Trudy convince Bob that she is Alice, that is, how can Trudy break the authentication?
Second and third messages are subject to attack. Trudy can acquire Alice's public key to open the second message because it is public, and do the same for message three using Bob's public key. Trudy can just sniff these messages and easily unmasked these messages.

b Can Trudy convince Bob that she is Alice and also determine the session key that Bob will use?
In this case, Trudy is not able to find $g^b \bmod p$, so Trudy cannot convince Bob that she is Alice.