

Submission Policy: Submissions must be typed (not handwritten) and stapled hardcopies are to be turned in (no softcopies). Only two late submissions (maximum delay from due date is "two days") per student are acceptable.

Problem: Read the section on "Key Exchange" on page 80-81-82 in your textbook (4th edition; 3rd edition also contains this section). S is the sender and R is the receiver. They do not know each other earlier. They want to talk secretly, which is why they want to agree on a session key K. The way they agree on the session key is given in this section, which is:

(A) $S \rightarrow R$ (means S sends to R): $E(K_{\text{pub-R}}, E(K_{\text{priv-S}}, K))$. Call this protocol as P1.

- (15 Points) If the encryption is changed as follows, which one(s) also would solve the above problem of session key Exchange? For each option answer **True**, if it solves, **False**, if it does not.

- $E(K_{\text{pub-S}}, E(K_{\text{priv-R}}, K))$ **FALSE**
- $E(K_{\text{pub-R}}, E(K_{\text{pub-S}}, K))$ **FALSE**
- $E(K_{\text{priv-R}}, E(K_{\text{priv-S}}, K))$ **FALSE**

- (15 Points) Let D be the decryption function (like E is the encryption function). Write just the formula using which R can find the session key K from the protocol P1.

Let $C = E(K_{\text{pub-R}}, E(K_{\text{priv-S}}, K))$.

$K = D(K_{\text{pub-S}}, D(K_{\text{priv-R}}, C))$. Look at the order of decryption: it is important.

- (10 points) Suppose after R gets K, she would send a message M to R using this key. Write the protocol using which R encrypts M.

Simple one is: $E(K, M)$; other one is $E(H(M), M)$. H is a crypto hash.

- (60 points) **Break P1**: Suppose you (i.e., Y) want to carry out a man-in-the-middle attack on P1. You would be happy if you can do some or all of the following: (a) carry out a replay attack, (b) carry out a man-in-middle attack such that you can learn message M that R sends to S. Show how you can carry out (a) and (b) in a step by step manner using the notation in (A), i.e., each step should be stated as: $S \rightarrow R: E(K_{\text{pub-R}}, E(K_{\text{priv-S}}, K))$.

- (i) $S \rightarrow R: E(K_{\text{pub-R}}, E(K_{\text{priv-S}}, K))$; Y records.
(ii) $R \rightarrow S: E(K, M)$; Y records.
Y can replay either (i) or (ii) or both. This is a simple attack. If it is a bank transaction, then it works (within the window during which K is valid).
- (i) $S \rightarrow R: E(K_{\text{pub-R}}, E(K_{\text{priv-S}}, K))$; Y intercepts and sends the message in (ii) to R instead.
(ii) $Y \rightarrow R: C = E(K_{\text{pub-R}}, E(K_{\text{priv-Y}}, K2))$; K2 is different than K.
(iii) R decrypts: $K3 = D(K_{\text{pub-S}}, D(K_{\text{priv-R}}, C))$; K2 is different than K.
Note that: $D(K_{\text{priv-R}}, C)$ gives $C3 = E(K_{\text{priv-Y}}, K2)$. Y also knows this.
Then R applies $D(K_{\text{pub-S}}, C3)$. This is K3. Since Y knows $K_{\text{pub-S}}$, it also computes K3.
(iv) Y in the meantime, computes: $K3 = D(K_{\text{pub-S}}, C3)$.
(v) $R \rightarrow S: E(K3, M)$. Y intercepts and decodes M.

5. **(Bonus: 100 Points)** Give a protocol that enhances the security of P1 by modifying it or proposing another protocol for secure session key exchange. You must prove informally that (a) and (b) in (4) cannot be carried out.
- In order to prevent (a) replay attacks, use timestamps. Let T be the timestamp.
 $S \rightarrow R: E(K_{\text{pub-R}}, E(K_{\text{priv-S}}, (T, K)))$.
 - In order to prevent the man-in-the middle attack: a simple solution is to use the Diffie Hellman protocol or the certificate C_s of each party, and a trusted third-party certificate authority.
 $S \rightarrow R: E(K_{\text{pub-R}}, (C_s, E(K_{\text{priv-S}}, (C_s, T, K))))$.
 R decrypts as in question 2, and matches the outer C_s and inner C_s . If they match, it is from S , else it is not and discard this message.