1. Describe in detail a man-in-the-middle attack on the Diffie-Hellman key-exchange protocol whereby the adversary ends up sharing a key \( k_A \) with Alice and a different key \( k_B \) with Bob, and Alice and Bob cannot detect that anything has gone wrong.

What happens if Alice and Bob try to detect the presence of a man-in-the-middle adversary by sending each other (encrypted) questions that only the other party would know how to answer?

2. Consider the following key-exchange protocol:

- **Common input:** The security parameter \( 1^n \).
- (a) Alice runs \( G(1^n) \) to obtain \((G, q, g)\).
- (b) Alice chooses \( x_1, x_2 \leftarrow Z_q \) and sends \( \alpha = x_1 + x_2 \) to Bob.
- (c) Bob chooses \( x_3 \leftarrow Z_q \) and sends \( h_2 = g^{x_3} \) to Alice.
- (d) Alice sends \( h_3 = g^{x_2 \cdot x_3} \) to Bob.
- (e) Alice outputs \( h_2^{x_1} \cdot h_3^{-1} \). Bob outputs \( (g^\alpha)^{x_3} \cdot (h_3)^{-1} \).

Show that Alice and Bob output the same key. Analyze the security of the scheme (i.e., either prove its security or show a concrete attack).

3. Show that any 2-round key-exchange protocol (that is, where each party sends a single message) can be converted into a CPA-secure public-key encryption scheme.

4. Consider the following variant of El Gamal encryption. Let \( p = 2q + 1 \), let \( G \) be the group of squares modulo \( p \), and let \( g \) be a generator of \( G \). The private key is \( (G, g, q, x) \) and the public key is \( G, g, q, h \), where \( h = g^x \) and \( x \in Z_q \) is chosen uniformly. To encrypt a message \( m \in Z_q \), choose a uniform \( r \in Z_q \), compute \( c_1 := g^r \ mod p \) and \( c_2 := h^r + m \ mod p \), and let the ciphertext be \( (c_1, c_2) \). Is this scheme CPA-secure? Prove your answer.

5. Consider the following modified version of padded RSA encryption: Assume messages to be encrypted have length exactly \( ||N||/2 \). To encrypt, first compute \( \hat{m} := 0x00 || r || 0x00 || m \) where \( r \) is a uniform string of length \( ||N||/2 - 16 \). Then compute the ciphertext \( c := [\hat{m}^d \ mod N] \). When decrypting a ciphertext \( c \), the receiver computes \( \hat{m} := [c^d \ mod N] \) and returns an error if \( \hat{m} \) does not consist of 0x00 followed by \( ||N||/2 - 16 \) arbitrary bits followed by 0x00. Show that this scheme is not CCA-secure. Why is it easier to construct a chosen-ciphertext attack on this scheme than on PKCS #1 v1.5?

6. In Section 12.4.1 we showed an attack on the plain RSA signature scheme in which an attacker forges a signature on an arbitrary message using two signing queries. Show how an attacker can forge a signature on an arbitrary message using a single signing query.