Public Key Encryption

1. In a public-key system using RSA, you intercept the ciphertext C = 10 sent to a user whose public key is e = 5, n = 35. What is the plaintext M?
2. Suppose Bob uses the RSA cryptosystem with a very large modulus n for which the factorization cannot be found in a reasonable amount of time. Suppose Alice sends a message to Bob by representing each alphabetic character as an integer between 0 and 25(A →0,..., Z→25), and then encrypting each number separately using RSA with large e and large n. Is this method secure? If not, describe the most efficient attack against this encryption method.
3. Users A and B use the Diffie-Hellman key exchange technique with a common prime q = 71 and a primitive root $\alpha = 7$.

   a. If user A has private key $X_A = 5$, what is A's public key Y_A?
   b. If user B has private key $X_B = 12$, what is B's public key Y_B?
   c. What is the shared secret key?

4. Is 3 a primitive root of 11? Why?
5. In an RSA system, the public key of a given user is e = 31, n = 3599. What is the private key of this user? Hint: You will need extended Euclidean algorithm to find the multiplicative inverse of 31 modulo $\phi(n)$.
6. True or False (and why?)
   a. Integrity can be achieved without message authentication.
   b. ECC can be used to provide confidentiality.
   c. For a public key system to work properly, it should not be possible (practically) to learn either of the two keys from each other.
   d. Man-In-The-Middle Attack can be used to defeat the security of Diffie-Hellman exchange.

1. In 1985, T. ElGamal announced a public-key scheme based on discrete logarithms. As with Diffie-Hellman, the global elements of the ElGamal scheme are a prime number q and α, a primitive root of q. A user A selects a private key X_A and calculates a public key Y_A as in Diffie-Hellman. User A encrypts a plaintext M < q intended for user B:
   1. Choose a random integer k such that $1 \leq k \leq q-1$.
   2. Compute $K = (Y_B)^k \bmod q$.
   3. Encrypt M as the pair of integers (C_1, C_2) where $C_1 = \alpha^k \bmod q$, $C_2 = KM \bmod q$

   User B recovers the plaintext as follows:

   1. Compute $K = (C_1)^{X_B} \bmod q$.
   2. Compute $M = (C_2K^1) \bmod q$.

   Show that the system works; that is, show that the decryption process does recover the plaintext.