Problem Set 5

Due: April 30

1. Stinson, problem 7.9, p.304. Only decrypt the first three ciphertext elements, namely (3781, 14409), (31552, 3930), and (27214, 15442).

2.

i. Let x be a positive integer. Suppose we're given that x is a divisor of 20, and that $x \neq 1$, $x \neq 2$, and $x \neq 5$. What are the possible values of x? [There are no tricks here; this problem is every bit as easy as it looks.]

ii. Let p and q be primes such that $p = 2q+1$. Let α be a random element of Z_p^*. Prove that if neither $\alpha^2 \bmod p$ nor $\alpha^q \bmod p$ is equal to 1, then α is a generator of Z_p^*.

[Hints: (1) For this problem you may use theorem 7.5 without proof. (2) There's a reason I asked you to do part (i) first.]

Note: Another fact stated in the notes is that the number of generators of Z_p^* is $\phi(p-1)$. In this case, we have $\phi(p-1) = \phi(2q) = \phi(2)\phi(q) = q-1$. Therefore, the probability that a randomly selected element of Z_p^* is a generator is about .50. So the fact I'm asking you to prove in this problem provides an efficient method for finding a generator of Z_p^*, as long as we can find a p and q of the required form. It turns out that there are reasonably efficient techniques for finding pairs of primes of this form.

3. Stinson, problem 5.12(a), p.181.

[Note: This problem is asking you to show that if h_1 is collision resistant, so is h_2. Stinson is suggesting that you prove the (logically equivalent) contrapositive: if h_2 is *not* collision resistant, then neither is h_1. To prove that h_1 is not collision resistant, it isn't enough to show that collisions *exist*. We need to show that we can efficiently *find* a collision for h_1, given a collision for h_2.]

4.

i. Suppose Bob uses ElGamal to encrypt two different messages to Alice, but carelessly uses the same random k (same ephemeral key) for both encryptions. Thus Bob creates the the ciphertexts,

   1. (γ, δ_1)
   2. (γ, δ_2)

   Suppose that you have intercepted both ciphertexts; know Alice's public parameter p; *and* have discovered the plaintext m_1 corresponding to the first ciphertext. Describe an algorithm for finding the second plaintext m_2.

ii. Apply the method described in part (i) to find the plaintext corresponding to the second ciphertext in the following example. The two ciphertexts are

   1. (1430, 697)
   2. (1430, 1113).

   You have intercepted the ciphertexts, know Alice's public parameter $p = 2357$, and have discovered that the plaintext corresponding to the first ciphertext (1430,697) is 2035.

5.

i. Suppose that Alice uses the ElGamal signature scheme to sign two different messages, m_1 and m_2. Her private key is a. As usual, the public parameters are (p, α, β), where α is a generator for Z_p^*, and $\beta \equiv \alpha^a \pmod{p}$. Suppose further that Alice carelessly uses the same ephemeral key (same value of k) for both signatures. Thus she constructs the signatures (r, s_1) and (r, s_2) for the two messages. Finally, assume that $gcd(s_1 - s_2, p - 1) = 1$. Show how Earl can discover the value of k efficiently in this case, given that he knows both m_1 and m_2 as well as the two signatures.

   [Hint: We have

$$s_1 = k^{-1}(h(m_1) - ar) \bmod (p-1) \tag{1}$$

$$s_2 = k^{-1}(h(m_2) - ar) \bmod (p-1) \tag{2}$$

   Subtract the second equation from the first.]

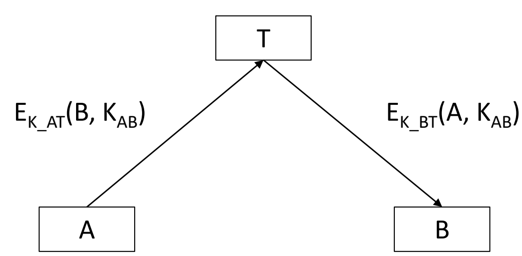
ii. Suppose $p = 31847, g = 5, \beta = 25703$, and that you intercept

a. The message $m_1 = 8990$ and corresponding ElGamal signature (23972,31396)

b. The message $m_2 = 31415$ and corresponding ElGamal signature (23972,20481)

Assume no hash function has been used, so that, for a message m, $s = k^{-1}(m - ar) \bmod (p-1)$. Find the ephemeral key k.

6.



The figure displays a simple (and very bad) key establishment protocol. Here T denotes the trusted authority Trent and, as usual, A is Alice and B is Bob. $E_{K_{AT}}$ and $E_{K_{BT}}$ are the symmetric keys that Alice and Bob respectively share with Trent. In the protocol, Alice sends Trent an encrypted message telling him that she wants to share the key K_{AB}, which she has created, with Bob. Trent decrypts, verifies that the message has the correct format, and sends the key to Bob. Trent's message to Bob includes a designator for Alice, so Bob knows that Trent intends for him to share the key with Alice.

Show how Earl can easily trick Bob into sharing a key with *him*, rather than with Alice. We assume (as is customary for protocol analysis) that the encryption is unbreakable.