Homework 3

Due Date: November 16, 2000 **200 Points**

1. (*30 points*) Chapter 9, exercise 2
2. (*30 points*) Chapter 9, exercise 16
3. (*10 points*) Chapter 9, exercise 18
4. (*10 points*) Chapter 9, exercise 19
5. (*60 points*) Consider double encryption, where $c = E_{k'}(E_k(m))$ and the keys k and k' are each n bits long. Assume each encipherment takes 1 time unit. A cryptanalyst will use a known plaintext attack to determine the key from two messages m_0, m_1 and their corresponding ciphertexts c_0 and c_1.
   a. The cryptographer computes $E_x(m_0)$ for each possible key x and stores each in a table. How many bits of memory does the table require? How many time units does computing the entry take?
   b. She then computes $y = D_{x'}(c_0)$, where D is the decipherment function corresponding to E, for each possible key x'. She then checks the table to see if y is in it.. If so, (x, x') is a candidate for the key pair. How should the table be organized to allow the cryptographer to find a match for y in time $O(1)$? How many time units would pass before a match must occur?
   c. How can the cryptographer confirm that (x, x') is in fact the key pair she seeks?
   d. What is the maximum time and memory needed for the attack? What is the expected time and memory?
6. (*20 points*) A network consists of n hosts. Assuming cryptographic keys are distributed on a per-host-pair basis, please compute how many different keys are required.
7. (*40 points*) Consider an RSA digital signature scheme. Alice tricks Bob into signing messages m_1 and m_2 such that $m = m_1 m_2 \bmod n_{Bob}$. Prove that Alice can forge Bob's signature on m.