Homework 2

1. Read 95-104, 109-114
2. (2 pts) Book, chapter 3, problem 1: come up with as efficient an encoding as possible to specify a completely general mapping between 64-bit inputs and 64-bit outputs.
3. (2 pts) Book, chapter 3, problem 3: How many DES keys, on average, encrypt a particular plaintext block to a particular ciphertext block?
4. (2 pts) Suppose instead of transmitting the displayed value on a token card to the server, the local computer saves the value and uses it as a shared-secret with the server in order to encrypt and integrity-protect the entire conversation. Assuming a machine that can execute 2 billion instructions per second, and assuming it takes 25 instructions to test a particular token value, how long would it take to break the session key? (hint: how many bits are in the displayed value, given each character has 16 possible values and there are 8 characters?)
5. (3 pts) Discuss at least three different types of token cards, and the special challenges involved in implementing each type, including cost (if they must have a keyboard), ability to have two-factor authentication, skew, etc.
6. (2 pts) Suppose it were illegal to use key sizes greater than 40 bits? How could you use an algorithm such as DES or AES and still manage to be legal?
7. (2 pts) Suppose you saw a ciphertext stream that you know was encrypted with CBC, and you saw that ci = cj? What information does this leak about the plaintext?
8. (3 pts) What is the time and memory requirements as compared to single DES for both the good guys and the bad guys, and explain the technique for doing brute force against double DES in each of the following cases?
   a) the same key is used twice (encrypt with K, then encrypt with K again)
   b) same key is used twice (first use K in encrypt mode, then use it in decrypt mode)
   c) assume two different keys (encrypt with K1, then encrypt with K2).
   d) encrypt with K1, decrypt with K2
9. (4 pts) Chapter 4, problem 4. Find a practical method of finding a triple of keys K1, K2, K3 that with triple DES map a particular input block to a particular output block. What is the expected time/memory? Why does this not imply triple DES is insecure? Can this same technique work if K1=K3?
10. (2 pts) How would you construct a message with a particular DES residue?
11. (1 pt) What would a Feistel cipher compute if the mangler function were f(x)=0, for all x?
12. (2 pts) Chapter 4 problem 2: Will K{IV} be the first block to repeat?