Problem 1: Overclockers Anonymous [5 pts]
Assume a cryptographic algorithm that is linear in the length of the key to perform “good guy operations”, e.g., encryption, decryption, key generation, integrity check generation, integrity check verification; and that it is exponential in the length of the key to perform “bad guy operations”, e.g., brute force breaking. Suppose advances in computation make computers an order of magnitude faster. Does this work to the advantage of the good guys, the bad guys, or neither? Justify your answer.

Problem 2. Public vs Private Keys [20 pts]
1. Assuming a very large message, and public user keys, describe what information would be included in each of the following scenarios. Explain efficiency issues, and alternate methods that would work but be less efficient.
   • Bob sending an unencrypted, signed message to Alice
   • Bob sending an unencrypted, signed message to multiple recipients (say Alice and Carol)
   • Bob sending an encrypted, signed message to Alice
   • Bob sending an encrypted, signed message to Alice and Carol

2. Assuming a very large message, and shared keys between each pair of users, describe what information would be included in each of the following scenarios. As with part 1, explain how the particular encoding you chose makes it efficient in terms of bandwidth and computation, and contrast it with alternatives.
   • Bob sending an unencrypted, integrity protected message to Alice
   • Bob sending an unencrypted, integrity protected message to Alice and Carol
   • Bob sending an encrypted, integrity protected message to Alice
   • Bob sending an encrypted, integrity-protected message to Alice and Carol

Problem 3. Hashtastic Functions [30 pts]
Bob has two hash functions, $f$ and $g$. He knows that one of them is collision-resistant (and the other isn’t), but he’s not sure which is which. He wants to create a new hash function $h$ which is definitely collision-resistant. Evaluate each of the following proposals, and either give a proof that it is definitely collision-resistant, or describe a counterexample (as usual, the $\circ$ symbol denotes concatenation):

1. $h(x) = f(x) \circ g(x)$
2. $h(x) = f(g(x))$
3. $h(x) = f(g(x)) \circ g(f(x))$
Problem 4. Two-Timing Pads [45 pts]

In this problem, we will explore why it’s never a good idea to reuse your one-time pads. I’ve provided six messages encrypted with three one-time pads (available on the course website).

The problem has three parts, each worth 15 points.

1. Discuss your approach to solving this problem. Consider the ASCII encoding system, the mathematical properties of exclusive-or, predictability of human and computer languages, and any other factors that will help you solve this problem.

2. Determine which pairs of ciphertext were encrypted using the same “one-time” pads.

3. Report the six plaintexts.

Hints:

• Each pad was used exactly twice.

• One of the messages is code, one of the messages is song lyrics, and one of the messages is a Shakespeare quotation.

• The following links may be helpful:
  – The NSA’s VENONA project: http://www.nsa.gov/public_info/declass/venona/