Math 456 (Washington) Final Exam May 17, 2006

Use 8 pages. For $i = 1$ to 8, solve problem i on page i.

1. (15 points) Describe how you would choose your plaintext for a chosen plaintext attack on each of the following and describe how you would use the information to obtain the key (do not write something like "choose some letters and look at what happens." Instead, you should write something like "let the plaintext be YTWNTRI and the key will be the first, third, and fifth letters of the resulting ciphertext"):

1. a Vigenère cipher.
2. an affine cipher
3. a Hill cipher with a 2×2 matrix

2. (14 points = 7+7) (a) Let $p = 17$ and let $s = 12$ be your secret. You want to share the secret among 3 people A, B, C so that any two can recover the secret, but no person can recover it alone. **Explicitly** list numbers you would give to each person in order to accomplish this.
(b) Let M be a long message, written as a binary string. Break M into 200-bit blocks $M_1||M_2||M_3 \cdots ||M_n$ (you may assume that the length of M is a multiple of 200). Let

$$H(M) = M_1 \oplus M_2 \oplus \cdots \oplus M_n.$$

What properties of a cryptographic hash function does H satisfy and which properties does it not satisfy?

3. (15 points =8+7) (a) Suppose you know that

$$33335^2 \equiv 670705093^2 \pmod{670726081}.$$

Describe how you can use this information to factor 670726081 (do not actually factor it; you may use the fact that 670726081 is a product of two primes).
(b) Suppose n is an odd positive integer. Let $k \equiv 2^{(n-1)/2} \pmod{n}$. Show that if $k^2 \not\equiv 1 \pmod{n}$, then n is not prime. Be sure to mention the name of any important theorem that you use.

4. (12 points = 7+5) (a) Let p be a large prime. Suppose you encrypt a message x by computing $y \equiv x^a \pmod{p}$ for some (suitably chosen) encryption exponent a. Let a_1 satisfy $a_1\, a \equiv 1 \pmod{p-1}$. Use Fermat's

theorem to show that $y^{a_1} \equiv x \pmod{p}$. You must explicitly show how Fermat's theorem is used. You may assume that $x \not\equiv 0 \pmod{p}$.
(b) Consider the following: Alice puts a message in a box and puts a lock on the box. Alice sends the box to Bob. Bob puts his lock on the box and sends the box back to Alice. Alice takes her lock off and sends the box back to Bob. Bob takes his lock off and reads the message.
Using the ideas of part (a), how would you implement this procedure mathematically?

5. (15 points = 5+5+5) (a) Let p be a large prime and let E be an elliptic curve mod p. Let Q be a point on E. Alice stores passwords in a file as follows. She represents the password as an integer k, computes $S = kQ$, and stores S in the file. When Bob logs onto the computer, he enters his password k_B, the computer computes k_BQ and compares the result with what is stored in the file under Bob's name.
(a) Suppose Eve gains access to the file. Why should it be difficult for Eve to obtain passwords.
(b) Suppose that Alice was lazy and chose Q to be the point at infinity. If Eve notices this, explain how Eve can log in easily as Bob.
(c) Describe a classical version of the procedure preceding (a) that is based on classical discrete logs.

6. (8 points) Let n be a product of two distinct large primes and let v be an integer with $\gcd(v, n) = 1$. Peggy claims to know a number s such that $s^2 \equiv v \pmod{n}$. Victor wants to verify this without determining the value of s. The first few steps of the procedure are as follows. Peggy chooses two random numbers r_1 and r_2 such that $r_1r_2 \equiv s \pmod{n}$, then computes $x_1 \equiv r_1^2 \pmod{n}$ and $x_2 \equiv r_2^2 \pmod{n}$. She sends x_1 and x_2 to Victor. Before proceeding, Victor of course checks that $v \equiv x_1x_2 \pmod{n}$. Describe a zero-knowledge verification that Peggy knows s, using these steps as the beginning of the procedure.

7. (6 points) Let E_K denote encryption (in some cryptosystem) using the key K. There are 2^{40} possible keys K. Suppose Alice uses the keys K_1 and K_2 and builds a double encryption machine that encrypts messages m to get ciphertexts $c = E_{K_1}(E_{K_2}(m))$. Suppose Eve gains access to Alice's machine. Describe a strategy that Eve can use to find possibilities for K_1 and K_2 in time significantly less than the 2^{80} steps that it takes to do a brute force search of all pairs (K_1, K_2).

8. (15 points = 5+5+5) Alice wants to sign a document m. She chooses

a cryptographic hash function H. She chooses an RSA modulus $n = pq$ (the product of two large primes) and exponents d and e satisfying $de \equiv 1 \pmod{(p-1)(q-1)}$. The function H and the numbers e and n are made public. To sign m, Alice lets

$$s \equiv H(m)^d \pmod{n}.$$

The signed document is (m, s). Bob verifies the signature by checking that $s^e \equiv H(m) \pmod{n}$.
(a) If Alice performs the signature steps correctly, why should the verification work?
(b) Suppose Eve chooses s_1 and tries to find m_1 such that (m_1, s_1) is valid. Why is this difficult?
(c) Suppose Eve instead chooses a document m_2 and tries to find s_2 such that (m_2, s_2) is valid. Why is this difficult? (*Hint:* It is equivalent to a certain decryption problem.)