UMBC
Fall 1994

CMSC–443
Cryptography and Data Security

Homework 5

Problem 1

Let n be any RSA modulus, and let e and d be any pair of corresponding encryption/decryption exponents. Let f_e denote RSA encryption under exponent e. For notational convenience, for any $x \in \mathbb{Z}_n$ let $[x]_n$ denote x mod n. Thus, for any $x \in \mathbb{Z}_n$, $f_e(x) = [x^e]_n$. Prove that RSA has the following multiplicativity and complementation properties:

1. For all messages $x, y \in \mathbb{Z}_n$, it is true that $f_e([xy]_n) = [f_e(x) f_e(y)]_n$.
2. For all messages $x, y \in \mathbb{Z}_n$, it is true that $f_e([n - x]_n) = [n - f_e(x)]_n$.

Problem 2

Design a simple mechanism that computes the "maximum" of two conventional cryptosystems. Describe your mechanism and prove that it has the desired property.

Motivation: Alice and Bob would like to communicate in private. Alice likes her crytosystem T_A; Bob likes his cryptosystem T_B. Because neither trusts the other, neither is willing to use the other's cryptosystem. Each would, however, be willing to use your mechanism that combines both cryptosystems, provided you can prove that the combined crytosystem is at least as hard to break as each of the two component systems.

Specifications: Your mechanism should take five inputs: T_A, T_B, k_A, k_B, and M, where k_A is a key for Alice's crytosystem T_A; k_B is a key for Bob's crytosystem T_B; and M is a message to be encrypted. From these inputs, your mechanism should compute a ciphertext C such that computing M from C requires breaking T_A and T_B. Describe how your mechanism works, and describe how the intended receiver can compute M from T_A, T_B, k_A, k_B, and C.

Hints: 1) Although it may be a good idea to compute the sequential combination $C = T_B T_A(M)$, no one has been able to prove that such a combination has the desired property. 2) Your mechanism is allowed to make us of a true random-number generator. 3) Think about the one-time pad, which computes $M \oplus r$, for a use-once key r. 4) Would it be helpful to combine hints (2) and (3)? In particular, would it be useful to generate two use-once random numbers r_A and r_B?