Computer Science 345
The Efficient Universe

Homework 9
Due Friday, May 5, 2006

**You may collaborate with other students,
but you should write up the solutions entirely on your own.**

Problem 1 (Merkle Key Exchange) Alice needs to send a secret message M to Bob. Unfortunately, an adversary, Eve, listens in on all communications between them. So Alice and Bob want to generate a secret key K which will be known only to them, but not to Eve. Then Alice and Bob can use a standard cryptosystem, for example, DES, to transfer the message M from Alice to Bob.

Let $f : S \to S$ be a one-way permutation, where S is a set of size n^2. For the purpose of this exercise, we assume that f is given as a black box or oracle. That is, Alice, Bob and Eve can call f as an external procedure, but they do not know how f works.

The Merkle Key Exchange is as follows:

1. Alice picks n random elements $x_1, \dots, x_n$ from the set S. Then she computes $y_1 = f(x_1), \dots, y_n = f(x_n)$ and sends $y_1, \dots, y_n$ to Bob.
2. Bob tries to guess one of x_i's: he picks a random $z \in S$ and computes $f(z)$.
   - if $f(z) = y_i$ for some i, then he sends i to Alice. Alice sets $K = x_i$; and Bob sets $K = z$ (note that $z = x_i$, since f is a permutation and $f(z) = y_i = f(x_i)$).
   - else Bob repeats step 2.

Remark: After the execution of the protocol, everybody (i.e. Alice, Bob and Eve) knows $y_1, \dots, y_n$ and i; Alice knows $x_1, \dots, x_n$; Bob knows $z = x_i$.

Prove that with probability 99% Bob needs only $O(n)$ calls to f to find z, and Eve needs $\Omega(n^2)$ calls to find the random variable K. (Note that the value of K depends on Bob: it is determined only after Bob guesses one of $x_1, \dots, x_n$.)

Problem 2 Let C be a circuit of depth d consisting of NAND gates. Show that there exists another circuit C' consisting of *unreliable* NAND gates of depth $O(d)$ computing the same function as C (every gate fails with probability at most ε, where ε is small; C' should output the correct answer with probability at least $2/3$). Find how small ε should be.

Remark: See Nick Pippenger's lecture for more details.
Recall, that the NAND function is defined as follows:

$$\mathrm{NAND}(x, y) = \neg(x \& y).$$

In other words,

$$\mathrm{NAND}(0,0) = \mathrm{NAND}(0,1) = \mathrm{NAND}(1,0) = 1;$$
$$\mathrm{NAND}(1,1) = 0.$$

Hint: Simulate "Majority of 3" by NAND gates.

Definition 1 *We say that a function f_K is a trapdoor one-way permutation with the public key $K \in \{0,1\}^n$ if*

1. *There exists an efficient algorithm G generating pairs of public and private keys (K, S). Both keys are binary strings of length n. Moreover, public keys K generated by G are distributed uniformly in the set $\{0,1\}^n$.*

2. *For a fixed $K \in \{0,1\}^n$, f_K is a permutation on the set $\{0,1\}^n$.*

3. *It is easy to compute f_K: there exists an efficient algorithm computing $f_K(x)$ (given x and K).*

4. *It is hard to invert f_K without knowing the private key S. Namely, for every polynomial $p(\cdot)$ and a probabilistic polynomial time algorithm (adversary) A, and for every* sufficiently large *number n (i.e. there exists N such that for every $n \geq N$)*

$$\Pr_{x,K \in \{0,1\}^n} (A(f_K(x)) = x) < 1/p(n).$$

5. *It is easy to invert (decrypt) f_K using the private key: there exists an efficient algorithm computing $f_K^{-1}(x)$ (given x, K and S).*

Remark: *This definition is a slightly simplified version of the standard definition.*

Problem 3 In this exercise we will construct an oblivious transfer protocol for *honest* (but curious) parties. Alice wants to send one of two messages M_1 or M_2 to Bob; Bob knows the index $b \in \{1, 2\}$ of the message he wants to receive. In the end of the protocol Bob should receive M_b; Alice should not know anything about b; Bob should not be able to reconstruct the other message. We assume that the parties are honest: that is they precisely follow the protocol. Consider the following protocol:

1. Bob generates a pair of public and private keys (K, S) of length n and a random binary string R of length n.

2. If $b = 1$, then Bob sends the pair $(P_1, P_2) = (K, R)$ to Alice. If $b = 2$, he sends $(P_1, P_2) = (R, K)$. Note that Alice receives two strings P_1 and P_2, not knowing the value of b.

3. Alice computes $F_1 = f_{P_1}(M_1)$ and $F_2 = f_{P_2}(M_2)$ and sends the strings to Bob.

4. If $b = 1$, then Bob reconstructs M_1 by computing $f_K^{-1}(M_1) \equiv f_{P_1}^{-1}(M_1)$. Note that he knows the private key S and thus can compute f_K^{-1} efficiently.

5. If $b = 2$, then Bob reconstructs M_2 by computing $f_K^{-1}(M_2) \equiv f_{P_2}^{-1}(M_2)$.

Show that Alice cannot guess the bit b with probability significantly larger than 1/2. Prove that Bob cannot reconstruct the message $M_{(2-b)}$ with non-negligible probability assuming that M_1 and M_2 are distributed uniformly and independently in $\{0,1\}^n$.

Hint: Show that if one of these statements is not true, then it is possible to break the trapdoor one-way permutation.

Bonus Problem Show that if Bob is not honest, then he can obtain both messages M_1 and M_2.