Problem Set 2

This problem set is due in class on **Thursday, March 7th.**

Problem 1. [Access control]

You have founded a hot new startup that is building a trendy new high-tech padlock. Your padlock comes with a factory-set combination, and a serial number printed on the back. However, one potential problem is that some customers will forget their combination, and when they call you up and give their serial number, you need to be able to tell them their combination. Moreover, you need to make sure that only the factory can do this: thieves should not be able to deduce the combination by looking at the serial number.

(a) You're going to sell a lot of locks. Describe a solution that does not require much storage at the factory: say, sub-linear in the number of locks sold. (The lower the storage needed, the better.)

(You may make reasonable computational assumptions, such as that there exist secure one-way functions.)

(b) You tried the above, but were forced to disclose in your business plan the risk that, if some researcher finds a constructive proof that $P = NP$, then your padlocks may become insecure. This got too many confused looks from the venture capitalists, so now the goal is to try to avoid this. Describe a solution that is unconditionally secure.

(You may not make any unproven computational assumptions.)

Problem 2. [Computational indistinguishability]

We define a distance measure on probability distributions as follows:

$$d_R(D, D') = \max_A |\Pr_{x \leftarrow D}[A(x) = 1] - \Pr_{x' \leftarrow D'}[A(x') = 1]|$$

where the maximum is taken over all algorithms running with resources R. Here R denotes a set of resources, which is used to restrict the adversary. For instance, R might list a bound on number of steps of computation, on queries to the oracle, on the amount of memory used, or something else. If algorithm A runs with resources R and A' with R', then let $R + R'$ denote the resources used by running first A then A'.

Note that D, D' are (t, e)-indistinguishable if and only if $d_t(D, D') \leq e$, so we can see that this distance measure is of fundamental interest in cryptography. This problem will ask you to study basic properties of this distance measure.

(a) [*Composition*] Let L be any (possibly randomized) algorithm running with resources R', and let $L(D)$ denote the distribution obtained by picking x from D and then outputting $L(x)$. Prove: $d_R(L(D), L(D')) \le d_{R+R'}(D, D')$.

(b) [*Triangle inequality*] Let D' be any distribution whatsoever. Prove: $d_R(D, D'') \le d_R(D, D') + d_R(D', D'')$.

(c) [*Multiple samples*] If D_1, D_2 are distributions, let $D_1 \times D_2$ denote the distribution on pairs whose first component is chosen according to D_1 and second component according to D_2 (independently), i.e., $\Pr_{D_1 \times D_2}[(x_1, x_2)] = \Pr_{D_1}[x_1] \times \Pr_{D_2}[x_2]$. Prove: $d_R(D \times D, D' \times D') \le 2 d_R(D, D')$.

(d) [*Information-theoretic security and the variation distance*] Recall that the variation distance is defined by $V(D_1, D_2) = \frac{1}{2} \sum_x |\Pr_{D_1}[x] - \Pr_{D_2}[x]|$. When used as a resource bound, let ∞ denote that there are no restrictions on the adversary (e.g., the algorithm can have unbounded running time).
Prove: $d_\infty(D, D') = V(D, D')$.

(e) [*Information-theoretic security*] Prove: $d_R(D, D') \le d_\infty(D, D')$.

(f) [*Conditioning*] If E is an event, let $D|E$ represent the distribution of the output of D conditioned on E occurring, i.e., $\Pr_{D|E}[x] = \Pr_D[x|E] = \Pr_D[x \text{ and } E]/\Pr[E]$. Let $\overline{E}$ denote the complementary event to E, i.e., that E does not occur.
Prove: $d_R(D, D') \le d_R(D|E, D'|E) + \Pr[\overline{E}]$.

(g) [*A XOR lemma*] If D_1, D_2 are distributions on k-bit strings, let $D_1 \oplus D_2$ denote the distribution on $x_1 \oplus x_2$ when x_1 is chosen from D_1 and x_2 is chosen from D_2 (independently). Here $\oplus$ denotes the xor operator. In other words, $\Pr_{D_1 \times D_2}[y] = \sum \Pr_{D_1}[x_1] \times \Pr_{D_2}[x_2]$, where the sum is taken over all pairs (x_1, x_2) such that $x_1 \oplus x_2 = y$. Let U denote the uniform distribution on k-bit strings.
Prove: $d_R(D \oplus D', U) \le 2 d_\infty(D, U) d_\infty(D', U)$.

Problem 3. [Symmetric-key Crypto in the Random Oracle Model]

Let $R : \{0,1\}^n \to \{0,1\}^{3n}$ be a random oracle. The following two parts are to be solved in the random oracle model, i.e., given access to R.

(a) [*One-way functions*] Show that there exists a $(t, 2t/2^n)$-secure one-way function in this model.

(Here the definition of security for a one-way function should be modified to allow the adversary to make queries to its oracle R: i.e., $f^R : X \to Y$ is a (t, p)-secure one-way function if $\Pr[f^R(A^R(f^R(x))) = f^R(x)] \le p$ for all adversaries A running in time at most t, where the probability is taken over the choice of x uniformly at random from X, over the choice of the random oracle R, and over the coin flips of A. Note that f is allowed to make queries to R during its execution, as is A; of course, the choice of the algorithm A itself must be made without knowledge of R.)

(b) [*Pseudorandom generators*] Show that there exists a $(t, t/2^n)$-secure pseudorandom generator, too, in this model.

(As before, the definition of security for a PRG should be modified to allow the adversary to make queries to the random oracle R, i.e., $G^R : X \to Y$ is a (t, ϵ)-secure PRG if $|\Pr[A^R(G^R(x)) = 1] - \Pr[A^R(y) = 1]| \leq \epsilon$ for all adversaries A running in time at most t, where the probabilities are taken over the choice of x uniformly at random from X, over the choice of y uniformly at random from Y, over the choice of the random oracle R, and over the coin flips of A. Note that the PRG G is allowed to query R during its execution.)

Problem 4. [P vs. NP]

This problem will ask you to provide evidence that finding some provably-secure symmetric-key cryptosystem is likely to be at least as hard as showing that $P \neq NP$.

(a) Suppose there exists a sequence $f_1, f_2, \ldots$ of one-way functions so that $f_n : \{0,1\}^n \to \{0,1\}^{q(n)}$ is $(t(n), p(n))$-secure, where $t(n)$ and $1/p(n)$ are super-polynomial functions of n and where f_n can be computed in time polynomial in n. Show that, in this case, $P \neq NP$.

(b) Suppose $G : \{0,1\}^k \to \{0,1\}^{2k}$ is a (t, e)-secure PRG. Show that G is a $(t, e + 2^{-k})$-secure one-way function.

(c) Suppose $F : \{0,1\}^k \times \{0,1\}^k \to \{0,1\}^k$ is a (t, q, e)-secure PRF, for some $q \geq 2$. Show that there exists a (t, e)-secure length-doubling PRG.

Hence conclude that if there exists any construction of OWF's, PRG's, or PRF's with super-polynomial security, then $P \neq NP$.

Remark. This provides evidence that finding provably-secure constructions of one-way functions or PRG's will be very difficult, since this crypto-design problem seems to inherit al the difficulties of the $P \stackrel{?}{=} NP$ question. (In fact, since we usually ask for security against randomized adversaries, it seems one must also resolve the question $BPP \stackrel{?}{=} NP$, which is at least as hard and possibly harder; moreover, even $BPP \neq NP$ does not seem to be sufficient for the existence of symmetric-key cryptography.) It seems the best we can hope for is that our constructions are proven secure under some plausible (but unproven) assumption, such as that factoring is hard. ■

Problem 5. [A length-expanding construction suggested in class]

Let $G : \{0,1\}^k \to \{0,1\}^{k+1}$ be a PRG. If y is a $k+1$-bit string, define $L(y)$ to be its left k bits and $R(y)$ to be its right k bits (note that they overlap in $k-1$ bits), and define $G_L(x) = L(G(x))$, $G_R(x) = R(G(x))$. Define $H : \{0,1\}^k \to \{0,1\}^{2k+2}$ by $H(x) = (G(G_L(x)), G(G_R(x)))$.

A salesperson claims if G is any (t, e)-secure PRG, then H is guaranteed to be a $(t/4, 8e)$-secure PRG, and offers to sell you the rights to this construction. Should you believe her security claims? Give a convincing argument: e.g., a proof or counterexample will do nicely.

Problem 6. [Pseudorandom functions and permutations]

(a) Show that applying two Feistel rounds does not yield a pseudorandom function. In other words, define $F_{f,f'}(L, R) = (L \oplus f(R), R \oplus f'(L \oplus F(R)))$ where the secret key f, f' holds two randomly chosen functions, and show that F is not a secure pseudorandom function.

(b) Give a secure pseudorandom function $F : K \times X \to X$ so that F_k is bijective for each $k \in K$, yet F is not a secure pseudorandom permutation. Be sure to prove that F is a pseudorandom function (by giving a security proof under some plausible assumptions) but not a pseudorandom permutation (by giving an attack).

In other words, you're showing a separation: F is intended to be secure against chosen-plaintext attack, but not against adaptive chosen-plaintext/ciphertext attack.