CS 127/CSCI E-127: Introduction to Cryptography

Problem Set 5

Assigned: Oct. 11, 2013 Due: Oct. 18, 2013 (5:00 PM)

Justify all of your answers. See the syllabus for collaboration and lateness policies. Submit solutions by email to `mbun@seas` (and please put the string "CS127PS5" somewhere in your subject line).

Problem 1. (More candidate one-way function families) Which of the following are likely to be one-way functions families? Justify your answers by either giving a polynomial-time adversary that inverts the function with nonnegligible probability or by showing that the function's one-wayness follows from the one-wayness of one of the candidates given in class.

a) $f_N : \mathbb{Z}_N \to \mathbb{Z}_N$ defined by $f_N(x) = [x^2 + 2x \bmod N]$, where $N = pq$ for random n-bit primes p, q.

b) $f_{p,x} : \mathbb{Z}_p^* \to \mathbb{Z}_p^*$ defined by $f_{p,x}(y) = y^x \bmod p$, where p is a random n-bit prime and $x \xleftarrow{\text{R}} \{0, \dots, p-2\}$.

Problem 2. (Modular exponentiation and hardcore bits) The fact that the least significant bit is not a hardcore bit for the modular exponentiation family ($f_{p,g}(x) = [g^x \bmod p]$) follows from the fact that x is even iff $f_{p,g}(x)^{(p-1)/2} \equiv 1 \bmod p$ (as discussed in section and §11.1.1 of KL 1st ed.). Show that the *second* least significant bit is also not a hardcore bit. You may use the fact that a random n-bit prime will be of the form $4k + 1$ for integer k with probability $\approx 1/2$.

Problem 3. (Bit-commitment schemes) A *bit-commitment scheme* is a cryptographic primitive that involves two parties, a *sender* and a *receiver*. The sender *commits* to a value $b \in \{0, 1\}$ by sending the receiver a string (called the *commitment*). Later, the sender can "reveal" the value b by sending the receiver another string (called the *opening*), which the receiver checks against the commitment. The commitment should be (perfectly) *binding*, meaning that it should be impossible for the sender to open it as both a 0 and 1. On the other hand, the commitment should be (computationally) *hiding* in that the committed value should be completely hidden to a polynomial-time receiver prior to revelation.

a) Formally define the properties we want from a commitment scheme. (If you have trouble, then it may help to try part **??** first and then formalize the properties of the scheme you construct.)

b) Construct a commitment scheme from any one-way permutation (and hardcore bit).

c) Extra Credit: Construct a (statistically binding) commitment scheme from any pseudorandom generator with expansion $\ell(n) \geq 3n$. Your scheme will probably require an extra step, where the receiver selects a random initialization string s which it sends to the sender, and the binding property will only hold with high probability over the receiver's choice of s. (Hint: Make use of $G_s(x) = G(x) \oplus s$ in addition to G itself.)