Handout 11: Problem Set 6

Instructor: Anna Lysyanskaya *Teaching Assistant: Moses Liskov*

This problem set is due on Wednesday, November 7.

Problem 1

(24 points) Recall that in lecture we mentioned that pseudorandom generators come in handy when we want to reduce the communication complexity (i.e., length of the data sent) of encryption. This problem asks you to give a proof for this.

(a) Let g be a pseudorandom number generator, from k bits to ℓ bits. Suppose a semantically secure cryptosystem (G, E, D) is given. In this cryptosystem, encrypting a message of size n results in communication complexity $p(n)$. Suppose Alice's public key is PK, which was obtained with security parameter k. Bob wants to send to Alice an ℓ-bit message m. To do so, he proceeds as follows: he creates a random k-bit seed s. From that, he sets his pseudo-one-time-pad $r = g(s)$. The ciphertext he then sends is $(m \oplus r, E(s))$. Is the resulting encryption scheme semantically secure? Prove your answer.

(b) Suppose that, instead, Alice's PK is a public key of a trapdoor permutation P_{PK}. In order to send m, Bob comes up with a random k-bit seed s, and computes $r = g(s)$. He then sends $(m \oplus r, P_{PK}(s))$. Is the resulting encryption scheme semantically secure? Prove your answer.

(c) Consider the following cryptosystem:

Key generation: generate $n = pq$ where p and q are primes of length k such that $p \equiv q \equiv 7 \bmod 8$. Public key is n, secret key is p, q.

Encryption: to encrypt an ℓ-bit message m, choose a random $s \in \mathbb{Z}_n^*$, create a random pad r as follows: $x_1 = s^2 \bmod n$; $x_i = x_{i-1}^2 \bmod n$ for $2 \leq i \leq \ell + 1$; $r_i = LSB(x_i)$ for $1 \leq i \leq \ell$. Output $(m \oplus r, x_{\ell+1})$.

First, describe how to decrypt in this scheme with two modular exponentiations and $O(\ell)$ modular multiplications. *Hint: The fact that* $p \equiv 7 \bmod 8$ *comes in handy.* Second, formulate a hardness assumption that corresponds to factoring n of the form used in this system, and prove that this cryptosystem is GM-secure under your assumption.

Problem 2

(21 points) A message authentication code (MAC) is a method for ensuring that the data received over the network came from the right person. More precisely, it is an object that satisfies the following properties: Suppose Alice and Bob share a secret s, and Bob wants to send message m to Alice. Then $MAC_s(m)$ is an efficiently computable function such that

even if an active attacker Eve queries it on a set of messages of its choice, it still cannot authenticate any message not explicitly queried. To give a more formal definition, let us first extend our notation. If $A^?$ is an oracle Turing machine, then by $Q \leftarrow A^O(x)$ we denote the contents of A's query tape upon termination with oracle O and input x.[1] Now: A function family $\{M_s(\circ)\}$ is a MAC if for all PPTF $\{A_n\}$, there exists a negligible function $\nu(k)$ such that

$$\Pr[s \leftarrow \{0,1\}^k; (m,x), Q \leftarrow A_k^{M_s} : x = M_s(m) \text{ and } (m,x) \notin Q] \leq \nu(k)$$

That is to say the adversary cannot find a pair (m, x) such that $x = M_s(m)$ even with oracle access to M_s, so long as M_s does not ask the oracle for $M_s(m)$ and receive the answer x.

(a) Let $\{F_s : \{0,1\}^{|s|} \mapsto \{0,1\}^{|s|}\}$ be a PRF family. Show that it is a MAC.

(b) Is it the case that a MAC is a PRF? Prove your answer.

(c) Is it true that if MACs exist then PRFs exist? Prove your answer.

Problem 3

(20 points) In this problem, we will try to relate various kinds of assumptions about one-way functions to each other.

(a) Prove that if one-way function families exist, then one-way functions exist. Can your proof be adjusted easily to go through for OWPs? Informally, argue why or why not.

(b) A function is length-preserving if the length of its output is always equal to the length of its input. Prove that if one-way functions exist, then length-preserving one-way functions exist.

Problem 4

(25 points) Suppose that $F_s(x)$ is a pseudorandom function from k bit input to k bit output, where s is the seed and x is the input, decide whether each of the following are always pseudorandom functions. If so, prove it. If not, provide a counterexample and explain how to distinguish the given function from random. It is okay to leave out the details of your proofs so long as they can be fleshed out correctly without too much trouble.

(a) $F_s^1(x) = F_{0^k}(x) \circ F_s(0^k)$

(b) $F_s^2(x) = F_{0^k}(s) \circ F_s(x)$

(c) $F_s^3(x) = F_s(x) \circ F_x(s)$

(d) $F_s^4(x) = F_{s_1}(x) \circ F_{s_2}(x)$ where $s_1 = F_s(x)$ and $s_2 = F_{\overline{s}}(x)$

(e) $F_s^5(x) = F_s(F_x(x))$

[1] By the query tape, we mean the list of all oracle queries and the answers that were returned in the course of A's execution.

Problem 5

(10 points) Summarize what we know so far. We have discussed several primitives in this class: secure public-key cryptosystems (PKC), one-way functions and one-way function families (OWF and OWFF), one-way permutations and one-way permutation families (OWP and OWPF), pseudorandom generators (PRG), pseudorandom functions (PRF), hardcore bits (HB), and trapdoor predicates (TP). For each pair of these, specify whether we know if the existence of one implies the existence of the other. Give as many such inferences as you can, and for each inference give either a reference to a paper, book, or problem set problem, or give a brief proof idea. *Since implications are transitive, you do not need to give every arrow you can deduce - just give as many independent results as you can.*