Introduction to Cryptography and Software Security
Problem Set 1: Private-Key Encryption

Due date: 6/4/2017

Instructions.

- You are allowed to rely on any statement that we proved in class, unless you are explicitly asked to prove it, and as long as you state it clearly and accurately.
- Justify your answers with formal proofs.
- **On-line submissions only! Hard copies will not be accepted.** Please make sure that scanned submissions are readable (unreadable submissions will not be accepted).

Problem 1 (0%). Let $\Pi = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be a symmetric-key encryption scheme. Prove that Π is *perfectly secret* if and only if for any message distribution M over $\mathcal{M}$, for any message $m \in \mathcal{M}$ such that $\Pr[M = m] > 0$, and for any ciphertext $c \in \mathcal{C}$ it holds that

$$\Pr[C = c | M = m] = \Pr[C = c] \,.$$

Problem 2 (0%). Prove that the one-time pad has indistinguishable encryptions. Does any perfectly-secret encryption scheme have indistinguishable encryptions?

Problem 3 (0%). Prove that for any negligible function $\nu(n)$ and for any polynomial $p(n)$ the function $p(n) \cdot \nu(n)$ is negligible.

Problem 4 (20%). Let G_1 and G_2 be pseudorandom generators.

1. Define $G(s) = G_1(s)||G_2(s)$ where $||$ denotes the concatenation of two strings. Prove that G is not necessarily a pseudorandom generator.

   [Instructions: Assume the existence of an arbitrary PRG, and construct G_1 and G_2 such that: (1) G_1 and G_2 are PRGs, but (2) G defined by G_1 and G_2 as above is not a PRG.]

2. Define $G(s_1||s_2) = G_1(s_1)||G_2(s_2)$ where $|s_1| \in \{|s_2|, |s_2| + 1\}$.[1] Prove that G is a pseudorandom generator.

Problem 5 (10%). Assuming the existence of a length-doubling pseudorandom generator G, construct length-doubling pseudorandom generator H such that $H(0^n) = 0^{2n}$ for every $n \in \mathbb{N}$.

Problem 6 (10%). Let F be a pseudorandom function, and consider the function $G : \{0,1\}^* \to \{0,1\}^*$ defined as

$$G(s) = F_s(1)||F_s(2)||\cdots||F_s(n+1)$$

where $n = |s|$ (and we naturally view the integers $1, \ldots, n+1$ as elements of $\{0,1\}^*$ via their binary representation). Prove that G is a pseudorandom generator.

[1] That is, for a input of length n bits we denote by s_1 its leftmost $\lceil n/2 \rceil$ bits and by s_2 its rightmost $\lfloor n/2 \rfloor$ bits.

Problem 7 (20%). Let F be a pseudorandom function such that for any $n \in \mathbb{N}$ and key $k \in \{0,1\}^n$ it holds that $F_k : \{0,1\}^n \to \{0,1\}^n$.

1. Let $P_k(x) = F_k(x) \oplus F_k(1^n)$ for any $n \in \mathbb{N}$, $k \in \{0,1\}^n$ and $x \in \{0,1\}^n$. Is P necessarily a pseudorandom function? Prove your answer.

   [Instructions: If your answer is YES, then prove that P is a PRF assuming that F is a PRF. If your answer is NO, then assume the existence of an arbitrary PRF, and construct a function F such that: (1) F is a PRF, but (2) P defined by F as above is not a PRF.]

2. Let $H_k(x) = F_k(x) \oplus 1^n$ for any $n \in \mathbb{N}$, $k \in \{0,1\}^n$ and $x \in \{0,1\}^n$. Is H necessarily a pseudorandom function? Prove your answer.

Problem 8 (20%). Let F be a pseudorandom function such that for any $n \in \mathbb{N}$ and key $k \in \{0,1\}^n$ it holds that $F_k : \{0,1\}^n \to \{0,1\}^{2n}$.

1. Let $H_k(x) = F_k(x) \oplus F_{1^n}(x)$ for any $n \in \mathbb{N}$, $k \in \{0,1\}^n$ and $x \in \{0,1\}^n$. Is H necessarily a pseudorandom function? Prove your answer.

2. Let $G(s) = F_{0^n}(s)$ for any $n \in \mathbb{N}$ and $s \in \{0,1\}^n$. Is G necessarily a pseudorandom generator? Prove your answer.

Problem 9 (20%). Let F be a pseudorandom function and let G be a pseudorandom generator with expansion $\ell(n) = n + 1$. For each of the following candidate encryption schemes state whether it is an IND-secure scheme or a CPA-secure scheme (and formally justify your answers). In all cases the encryption key is a uniformly sampled $k \in \{0,1\}^n$.

1. To encrypt a message $m \in \{0,1\}^{n+1}$, sample $r \leftarrow \{0,1\}^n$, and output the pair $(r, G(r) \oplus m)$.

2. To encrypt a message $m \in \{0,1\}^n$ output $F_k(0^n) \oplus m$.

3. To encrypt a message $m \in \{0,1\}^n$, sample $r \leftarrow \{0,1\}^n$, and output the pair $(r, r \oplus F_k(r) \oplus m)$.