Problem 1-1 (Key Recovery vs. Message Privacy?) 8 Points

Another definition of security that is sometimes (mistakenly) considered for private-key encryption is security against key-recovery attacks. (Indeed, in toy examples of insecure schemes we would always recover the secret key \( s \).)

Consider the following definition of this notion: An encryption scheme \((\text{Enc}, \text{Dec})\) with associated key space \( K \) and message space \( M \) is perfectly-secure against key recovery if the following holds for any algorithm \( \text{Eve} \) and any distribution \( M \) over the message space \( M \):

\[
\Pr\left[ \text{Eve}(c) = s \mid s \leftarrow K; m \leftarrow M; c \leftarrow \text{Enc}_s(m) \right] \leq \frac{1}{|K|}
\]

Answer the following questions:

(a) (1 point) Explain the intuitive appeal of the above definition in your own words.

(b) (4 points) Show that the above definition is not necessary for perfectly-secure encryption. I.e., show that there exists an encryption scheme which is perfectly-secure under the definition we gave in class, but not perfectly-secure against key recovery.

(c) (3 points) Show that the above definition is not sufficient for perfectly-secure encryption. I.e., show that there exists an encryption scheme which is perfectly-secure against key recovery, but not perfectly-secure under the definition we gave in class.

(Hint: Something “stupid” works.)

Problem 1-2 \((t, \epsilon)-security\) 25 Points

Consider the following definition of security for any (deterministic) secret key encryption scheme. A scheme \((\text{Enc}, \text{Dec})\) is \((t, \epsilon)\)-secure if for any messages \( m_0, m_1 \in M \), and any adversary \( \text{Eve} \) running in time at most \( t \), it holds

\[
|\Pr_S[\text{Eve}(\text{Enc}_S(m_0)) = 1] - \Pr_S[\text{Eve}(\text{Enc}_S(m_1)) = 1]| \leq \epsilon
\]

According to the values of \( t \in [0, \infty] \) and \( \epsilon \in [0, 1] \) one can obtain different notions of security.
**Part I.** Here you will show that \((t,0)\)-security implies \(|K| \geq |M|\) for a very small (polynomial) value of \(t\).

(a) (3 points) Argue that \((\infty,0)\)-security implies the following definition of security. For any messages \(m_0, m_1 \in M\), any ciphertext \(c \in C\), it holds:

\[
\Pr_S[Enc_S(m_0) = c] = \Pr_S[Enc_S(m_1) = c]
\]

(Hint: For any ciphertext \(c\), define a specific adversary \(Eve_c\) such that Equation (2) directly follows from applying Equation (1) (with \(\epsilon = 0\)) to \(Eve_c\).)

(b) (4 points) Using part (a), prove that \((\infty,0)\)-security implies Shannon’s security. (Hint: Use the Bayes law in the same way as in the proof of one-time pad, and then use Equation (2).)

(c) (3 points) Part (b) and Shannon’s theorem prove that \((\infty,0)\)-security implies that \(|K| \geq |M|\). However, it turns out that \((t,0)\)-security already implies Shannon’s security (and, hence, \(|K| \geq |M|\)) for a relatively small value of \(t\). Trace your proof in part (a) to establish the smallest \(t\) for which the above holds.

**Part II.** Here you will show that \((\infty,\epsilon)\)-security implies \(|K| \geq |M|(1 - \epsilon)\).

(d) (4 points) Argue \((\infty,\epsilon)\)-security implies the following definition of security. Let \(M_1\) denote the uniform distribution over the message space \(M\). Then, for every message \(m_0 \in M\), any subset \(T \subseteq C\) of ciphertexts, it holds:

\[
\left| \Pr_S[Enc_S(m_0) \in T] - \Pr_{S,M_1}[Enc_S(M_1) \in T] \right| \leq \epsilon
\]

(Hint: Write \(\Pr_S[Enc_S(m_0) \in T] = \sum_{m_1} \frac{1}{|M|} \Pr_S[Enc_S(m_0) \in T]\), and then expand the second probability over all possible choices of \(m_1 \leftarrow M_1\). Finally, use Equation (1) for some particular \(Eve\) which depends on \(T\).)

(e) (2 points) Show that

\[
\Pr_{S,M_1}[Enc_S(M_1) \in T] = \sum_s \frac{|\{m_1 \in M : Enc_s(m_1) \in T\}|}{|M| \cdot |K|}
\]

(f) (4 points) Given any message \(m_0\), let \(T_0 = \{Enc_s(m_0) : s \in K\}\) be the set of all encryptions of \(m_0\). Applying Equation (3) and Equation (4) to \(T_0\) above, show

\[
\sum_{s \in K} \frac{|\{m_1 \in M : Enc_s(m_1) \in T_0\}|}{|K|} \geq (1 - \epsilon) \cdot |M|
\]

(Hint: What is \(\Pr_S[Enc_S(m_0) \in T_0]\)?)
(g) (4 points) Show that for any \( s \) and \( m_0 \),
\[
|\{m_1 \in M : Enc_s(m_1) \in T_0\}| \leq |K|
\]  
(Hint: Using uniqueness of decryption, show \( |\{m_1 \in M : Enc_s(m_1) \in T_0\}| \leq |T_0| \) first.)

(h) (1 points) Combine Equation (5) and Equation (6) from parts (f) and (g) to show that \( |K| \geq |M|(1 - \epsilon) \).

Problem 1-3 (One-Way Functions)  

Remember the Password Authentication system studied in the class, where the user \( U \) stores the “hash” \( y = f(x) \) of his password at the server \( S \), and later authenticates himself by presenting \( x \). We argue that the security of this system is equivalent to the one-wayness of \( f \). Assume now we have \( N = \text{poly}(k) \) users \( U_1 \ldots U_N \) in the system, and Eve breaks the scheme if she can recover a valid password of any one of the \( N \) users.

(a) (2 point) As a warm up, write a formal definition of security for a multiple user Password Authentication system which formalizes the informal description above. Namely, your definition should have the form “For any \( N = \text{poly}(k) \) and any PPT Eve, \( \text{Adv}(Eve) = \text{negl}(k) \), where \( \text{Adv}(Eve) \) is defined as ...”

(b) (5 points) Once you have defined this notion of multiple-user password authentication, prove that the natural generalization of the OWF-based scheme — where all users choose random passwords, and the same one-way function is used for all of them — is secure. Namely, show a reduction from any Eve breaking multiple-user security of the system to some inverter \( A \) for the original OWF. Make sure you formally define \( A \). (Hint: guess which user’s password will be broken by Eve.)

(c) (5 points) Assume two users \( U_1 \) and \( U_2 \) share the same first half of their passwords. Namely, if we write a password \( x = x_1|x_2 \), where \( x_1, x_2 \in \{0,1\}^{k/2} \), assume \( U_1 \) has password \( x = x_1|x_2 \), and \( U_2 \) has password \( x' = x_1|x'_2 \), where \( x_1, x_2, x'_2 \) are otherwise random and independent from each other. Assume further that \( U_2 \)’s password was compromised, so that Eve effectively learned the first half \( x_1 \) of \( U_1 \)’s password \( x \). Show that, in general, the security of \( U_1 \) might have been compromised as well. Namely, assuming one-way functions exist, construct a function \( f \) s.t.

1. \( f \) (built from some auxiliary OWF \( g \)) is a OWF (if \( g \) is a OWF); and
2. given \( x_1 \) and \( y = f(x_1|x_2) \), it is easy to recover \( x_2 \) (and, thus, \( x = x_1|x_2 \)). (Hint: Use some OWF \( g : \{0,1\}^{k/2} \rightarrow \{0,1\}^{k/2} \) and build a function \( f \) using \( g \).)

Problem 1-4 (Fun with One-Way Functions)  

Which of the following families of functions is a family of one-way functions? Of course, you may assume that RSA, factoring, squaring and modulo exponentiation are one-way. Justify your answers by providing either (1) a proof of security (then state from what assumption); or (2) an efficient attack explicitly breaking the proposed candidate (with non-negligible probability).
(a) (2 points) \( f_n(x) = x^3 \mod \varphi(n) \), where \( n = pq \) for two \( k \)-bit primes \( p \) and \( q \), and the input \( x \) comes from \( \mathbb{Z}_n^* \).

(b) (2 points) \( f(x) = x^2 \), where \( x \) is an integer in \( \mathbb{Z} \) (and, for security parameter \( k \), \( x \) is chosen at random from \([2^{k-1}, 2^k - 1]\)).

(c) (4 points) \( f(x_1, x_2) = g(x_1) \mid g(x_1 \oplus x_2) \), where \( g : \{0,1\}^k \rightarrow \{0,1\}^k \) is a OWF, and the inputs \( x_1, x_2 \) are in \( \{0,1\}^k \).

(d) (3 points) \( f(x_1|x_2) = g(x_1|x_2) \mid x_1 \), where \( g : \{0,1\}^k \rightarrow \{0,1\}^k \) is a OWF, \( x_1 \in \{0,1\}^{k-t} \), \( x_2 \in \{0,1\}^t \), and \( t = O(\log k) \). (I.e., one gives “for free” most of the input \( x \).)

(e)* (Extra Credit) (6 points) \( f_{p,g,y}(x_1, x_2) = g^{x_1}y^{x_2} \mod p \), where \( p \) is a random \( k \)-bit prime, \( g, y \in \mathbb{Z}_p^* \) and the input \( x_1, x_2 \) come from \( \mathbb{Z}_{p-1} \).

(Hint: Try to reduce from discrete log \( h_{p,g}(x) = g^x \mod p \), using your own challenge \( y = g^x \) as the second generator in \( f_{p,g,y} \).)