0368.4162: Introduction to Cryptography, Fall 2008 Instructor: Ran Canetti

Problem Set 4

February 24, 2009 Due: Thursday, February 26 12:00pm

The weight of this problem set towards the final grade will be twice the weight of each other problem set. (That is, the contribution of problem sets to the overall grade will be an average of five values. Each of the first three problem set contributes one value, and this one contributes two.) There are two reasons for this: First, this one is somewhat longer than the other ones. Second, I'd like to incentivize you to study this material at more depth, since it is the most relevant to many current research directions in cryptography.

1. A popular approach for encrypting long messages using public-key (asymmetric) encryption is to first encrypt a (relatively short) key k using asymmetric encryption, and then to encrypt the message using a symmetric encryption scheme with key k. (The main gain here is efficiency, since symmetric schemes are significantly faster than asymmetric schemes.) The goal of this exercise is to prove the security of this approach, called *hybrid encryption.*

   Let $E_a = (GEN_a, ENC_a, DEC_a)$ be an asymmetric encryption scheme, and let $E_s = (ENC_s, DEC_s)$ be a symmetric encryption scheme. Let $E_h = (GEN_h, ENC_h, DEC_h)$ be the following asymmetric encryption scheme:

   - $GEN_h = GEN_a$
   - $ENC_h(ek, m)$ first chooses an n-bit random value k (where $n = |ek|$), and then outputs c_1, c_2 where $c_1 = ENC_a(ek, k, r_a)$, $c_2 = ENC_s(k, m, r_s)$, and r_a, r_s are random inputs for ENC_a and ENC_s, respectively.
   - $DEC_h(dk, c = (c_1, c_2)) = DEC_s(DEC_a(dk, c_1), c_2)$

   (a) **[20 points]** Show that if E_a and E_s are a CPA secure than E_h is CPA secure.

   (b) **[10 points]** Assume that E_a is CCA secure and E_s is CPA secure. Is E_h CCA secure? (Either prove or show a counter example.)

   (c) **[Bonus: 20 points]** Let $M = (AUTH, VER)$ be a secure MAC scheme, and consider the following variant of E_h, denoted $E'_h = (GEN'_h, ENC'_h, DEC'_h)$:

   - $GEN'_h = GEN_a$
   - $ENC_h(ek, m)$ first chooses two n-bit random values k_e, k_a, and then outputs c_1, c_2, t where $c_1 = ENC_a(ek, (k_e, k_a), r_a)$, $c_2 = ENC_s(k_e, m, r_s)$, $t = AUTH(k_a, c_2)$, and r_a, r_s are random inputs for ENC_a and ENC_s, respectively.
   - $DEC_h(dk, c = (c_1, c_2, t))$: let $(k_e, k_a) = DEC_a(dk, c_1)$, $m = DEC_s(k_e, c_2)$; if $VER(k_a, c_2) = 1$ then output m; else output $\perp$.

   Assume that E_a is CCA secure, E_s is CPA secure, and M is a secure MAC scheme. Is E'_h CCA secure? If so, prove it. Else, show a counter example and then show how to modify E'_h in a minimal way so that it becomes CCA secure.

2. Read Section 4.4.2 in Goldreich's book (Volume I). Then, consider the Blum protocol for Graph Hamiltonicity (i.e., the protocol we saw in class), where the commitments are instantiated with Pedersen commitments over an ensemble $G = \{G_n\}_{n \in \mathbf{N}}$ of groups, and where the basic three-message protocol is repeated sequentially n times, n is the input length, and the verifier accepts only if all n repetitions accept.

   (a) **[15 points]** Show that this protocol is perfect zero knowledge.

   (b) **[15 points]** Show that the protocol is computationally sound with negligible soundness error, under the Discrete Log assumption in G.

(c) **[Bonus: 15 points]** Show that the protocol is a proof of knowledge, under the Discrete Log assumption in G.

3. Coin-tossing is a task where two mutually distrustful parties wish to agree on a random unbiased bit. We'll consider two definitions of this task. In both definitions we're concerned with a two-party protocol $\pi = (\pi_1, \pi_2)$ were the possible local outputs for each party are $\{0, 1, \perp\}$, where $\perp$ represents an aborted execution with no output. Next:

   Game based: A protocol $\pi = (\pi_1, \pi_2)$ is a GB-coin-tossing protocol if

   $$\text{Prob}[[\pi_1, \pi_2](1^n, 1^n) = (b, b) \text{ for some } b \in \{0, 1\}] > 1 - \nu(n)$$

   and in addition, for any value $b \in \{0, 1\}$ and any polytime adversary A we have:

   $$\text{Prob}[[\pi_1, A](1^n, 1^n)_1 = b] < 1/2 + \nu(n)$$

   and

   $$\text{Prob}[[A, \pi_2](1^n, 1^n)_2 = b] < 1/2 + \nu(n)$$

   for some negligible function $\nu(n)$.

   Ideal-Model based: A protocol $\pi = (\pi_1, \pi_2)$ is an IM-coin-tossing protocol if π securely realizes the two-party randomized function $f(1^n, 1^n) = (b, b)$, where b is chosen uniformly from $\{0, 1\}$.

   (a) **[15 points]** Show that a protocol is a GB-coin-tossing protocol if and only if it is an IM-coin-tossing protocol.

   (b) **[15 points]** Show that if there exist one way permutations then there exist an IM-coin-tossing (or, equivalently, GB-coin-tossing) protocol.

   (Hint: Consider the following protocol outline. The first party commits to a random bit, the second party sends another random bit, then the first party opens its commitment and both parties output the xor of the two bits. To complete this sketch to a fully specified protocol, you need to fix a commitment scheme, and also to specify what each party does if the other deviates from the protocol in each step.)

   (c) **[10 points]** Formulate an extension of one of the the above definitions to the case where the parties want to output a joint n-bit random string.

   (d) **[Bonus: 20 points]** Extend the above protocol to a protocol where parties jointly choose an unbiased random string, and show that the protocol satisfies the definition of security that you formulated in (3c).

   (Extra points will be given to protocols that have only a constant number of rounds. This is not an easy question!)