University of Maryland
CMSC858K — Cryptography
Professor Jonathan Katz

Problem Set 6

Due at *beginning* of class on May 13

1. In class we stated that for the case of *public-key* encryption schemes, security against ciphertext-only attacks and security against chosen-plaintext attacks are equivalent. Prove this formally. *Hint*: you will need to use a hybrid-type argument here.

2. Consider the (non-standard) security notion of *random-message indistinguishability* (RND-IND). Informally, the definition says that the encryption of a random message r is "secure"; more precisely, if r, s are two random strings then $(\mathcal{E}_{PK}(r), r)$ is computationally indistinguishable from $(\mathcal{E}_{PK}(r), s)$. More formally, public-key encryption scheme $(\mathcal{G}, \mathcal{E}, \mathcal{D})$ is secure in the sense of RND-IND if the following is negligible for all PPT algorithms A:

$$\Big|\Pr[(PK, SK) \leftarrow \mathcal{G}(1^k); r \leftarrow \{0,1\}^k; C \leftarrow \mathcal{E}_{PK}(r) : A(PK, C, r) = 1] - \Pr[(PK, SK) \leftarrow \mathcal{G}(1^k); r, s \leftarrow \{0,1\}^k; C \leftarrow \mathcal{E}_{PK}(r) : A(PK, C, s) = 1]\Big|.$$

   (a) Assuming there exists a public-key encryption scheme secure in the sense of left-or-right indistinguishability (i.e., the definition given in class, which we abbreviate as STD-IND), show that there exists an encryption scheme $(\mathcal{G}, \mathcal{E}, \mathcal{D})$ which is secure in the sense of RND-IND but which is *not* secure in the sense of STD-IND.

   (b) Given a public-key encryption scheme $(\mathcal{G}, \mathcal{E}, \mathcal{D})$ which is secure in the sense of RND-IND, show how to construct a (simple) public-key encryption scheme $(\mathcal{G}', \mathcal{E}', \mathcal{D}')$ which is secure in the sense of STD-IND. Prove the security of your construction.

3. Consider the following modification of the El Gamal encryption scheme over cyclic group G of prime order q: the public key is (g, h), the secret key is $\log_g h$, and message $m \in \mathbb{Z}_q$ is encrypted by choosing random r and sending $(g^r, h^r g^m)$.

   (a) Show how the receiver can recover g^m.

   (b) If the discrete logarithm problem is hard in G, recovering g^m will not, in general, allow the receiver to efficiently recover m. But if we assume the sender only sends messages $m \in \{0, \ldots, 100\}$, then the receiver *can* recover m (how?). However, does the scheme remain "secure" if we restrict m in this way?

   (c) Say (A_1, B_1) is an encryption of m_1. Prove that $(A_1, B_1 \cdot g^{m_2})$ is an encryption of $(m_1 + m_2) \bmod |G|$.

   (d) Say (A_1, B_1) is an encryption of m_1 and (A_2, B_2) is an encryption of m_2. What is $(A_1 A_2, B_1 B_2)$ an encryption of?

(e) Assume the receiver R is conducting an auction in which two bidders each encrypt their bids and send them to R. The bid of the first bidder is assumed to be in the range $\{0, \ldots, 100\}$. Argue that the bidder who goes second can cheat and always win by bidding \$1 more than the first bidder *even without ever learning the value of the first bidder's bid.*

4. In class we suggested the following protocol for two parties A and B who are trying to flip an unbiased coin: (1) A commits to a random bit b_A and sends the resulting commitment to B. (2) B sends a bit b_B. (3) A opens its commitment and reveals b_A (if A is caught cheating here, B simply aborts). The value of the coin is $b = b_A \oplus b_B$.

   Consider the following modification of the above protocol: (1) A commits to a random bit b_A and sends the resulting commitment to B. (2) B commits to a random bit b_B and sends the resulting commitment to A. (3) A opens its commitment and reveals b_A (if A is caught cheating here, B simply aborts). (4) B opens its commitment and reveals b_B (if B is caught cheating here, A simply aborts). The value of the coin is $b = b_A \oplus b_B$.

   (a) Argue that a dishonest A cannot bias the result of the coin in this modified protocol.

   (b) Let g, h be two (public) generators of a finite, cyclic group G of prime order q. Consider the following commitment scheme for 1 bit: choose a random $x \in \mathbb{Z}_q$. To commit to a "0", compute $C = g^x$. To commit to a "1", compute $C = g^x h$. Prove that this commitment scheme perfectly hides the value of the committed bit (i.e., even from an all powerful receiver) and that the scheme is computationally binding for the sender.

   (c) Show that if this commitment scheme is used for the modified coin-flipping protocol above (where g, h are public and neither party knows $\log_g h$), then a dishonest B *can* bias the result of the coin and thus the protocol is not secure.

Class Survey

I would appreciate it if you would fill this out and return it with the homework; you may fill it out anonymously if you like. Please note that this survey is independent of the department's course evaluation (which I hope you will fill out as well).

(*Note*: I am also happy to get more detailed feedback on any of the questions.)

1. Rate the overall level of difficulty of the class:

   (too easy) 1 2 3 4 5 (too hard)

2. Rate the overall difficulty of the homeworks:

   (too easy) 1 2 3 4 5 (too hard)

3. How interesting did you find the selection of material, including the level of formality (proofs, etc.) at which it was taught?

   (boring) 1 2 3 4 5 (very interesting)

4. How often did you consult the web-based textbooks (e.g., Goldwasser-Bellare or Bellare-Rogaway)?

   (what textbooks?) 1 2 3 4 5 (very frequently)

5. How often did you consult research papers that were referenced from the class syllabus?

   (what references?) 1 2 3 4 5 (very frequently)

6. **Important:** How likely would *you* be to take an advanced cryptography course in the Spring, 2004 semester? This course would assume some previous familiarity with cryptography and provable security, and would be a seminar course in which students present papers. (Responses to this question will be used to determine whether such a class will be offered.)

   (definitely not) 1 2 3 4 5 (definitely yes)