0368.3049 **Fall 2013/14**

Introduction to Modern Cryptography

Benny Chor and Nir Bitansky

Assignment 4

Published January 9, 2013. Due January 16, in mailbox 372 (Schreiber building).
Submission in pairs is encouraged (submission in threes or more is not allowed). A 5-point bonus will be given to typed (as opposed to handwritten) submissions.

Remark: As usual, in all questions, assume that algorithms may be probabilistic, and in all probability statements, the probability is also taken over their random coin tosses. For $n \in \mathbb{N}$, we denote by $[n]$ the set of integers $\{1, \ldots, n\}$.

Problem 1: Zero-knowledge for Quadratic-Residousity. Let $N = pq$ be a product of two primes, and let $\mathbb{QR} = \left\{r^2 : r \in \mathbb{Z}_N^*\right\}$ denote the subgroup of quadratic residues in $\mathbb{Z}_N^*$. Consider the following protocol for proving quadratic-residousity.

$(P(x), V)(y)$

Common input: $y \in \mathbb{QR}$.
Private input of P: x such that $y = x^2 \mod N$.

$P \to V$: P samples a uniformly random $r \leftarrow \mathbb{Z}_N^*$, and sends $z = r^2 \mod N$ to V.
$P \leftarrow V$: V samples a uniformly random bit $b \leftarrow \{0,1\}$, and sends b to P.
$P \to V$: If $b = 0$, P sends $a_0 = r$ to V. If $b = 1$, P sends $a_1 = xr \mod N$ to V.
If $b = 0$, V accepts iff $a_0^2 = z \mod N$. If $b = 1$, V accepts iff $a_1^2 = zy \mod N$.

Figure 1: A protocol for proving quadratic residousity.

(a) **Soundness (15 points):** Show that, for common input $y \notin \mathbb{QR}$, any (even unbounded) prover P^*, fails to make V accept with probability at least $1/2$.

(b) **Zero-knowledge against honest verifiers (15 points):** Show that you can efficiently and perfectly simulate the view of an honest verifier. Concretely, show that there exists a polytime algorithm $S(y, b)$ that given $y \in \mathbb{QR}$, and $b \in \{0,1\}$, efficiently samples a first message $\tilde{z}$ and a third message $\tilde{a}_b$, such that $(\tilde{z}, b, \tilde{a}_b)$ has the exact same distribution as the messages (z, b, a_b) produced in a real execution of the protocol, where V uses the coin b.

(c) **Bonus - Zero-knowledge against malicious verifiers (5 points):** Show that you can efficiently and perfectly simulate the view of a malicious verifier V^*. Concretely, show that there exists an expected polytime algorithm $S(y)$ that given $y \in \mathbb{QR}$, efficiently samples all messages $(\tilde{z}, \tilde{b}, \tilde{a}_b)$ so that they have the exact same distribution as the messages (z, b, a_b) produced in a real execution of the protocol.

Problem 2: Threshold El-Gamal encryption. Recall the El Gamal public-key encryption system presented in the recitation, over a cyclic group $\mathbb{G}$ of prime order q, where the secret key is a uniformly random $x \leftarrow \mathbb{Z}_q$, the public key is of the form g, g^x (for some generator g of $\mathbb{G}$), and an

encryption of $m \in \{0,1\}$, using randomness $y \leftarrow \mathbb{Z}_q$, is of the form $E_{g,g^x}(m;y) = (g^y, g^{xy}g^m)$.

(a) (20 points) We wish to allow the owner of the secret key x to delegate decryption to his n sons, by giving them each a share sk_i of the key. It is required that, for each and every encryption, decryption is possible only if **all** n sons are involved in the process. Specifically, to decrypt a cipher c, each son i computes using c and sk_i a *c-designated* decryption key $sk_{i,c}$, such that given all $\{sk_{i,c}\}_{i\in[n]}$, it is possible to decrypt c. Any subset of sons $S \subsetneq [n]$ should not be able to break the semantic security of the encryption, even given their shares $\{sk_i\}_{i\in S}$. Furthermore, the decryption values $\{sk_{i,c}\}_{i\in[n]}$ for any given cipher c, should not break the security of a new independent cipher c'.

Describe how the El-Gamal decryption procedure can be extended to meet this requirement (the encryption algorithm should stay the same). There is no need to prove security, but only describe the construction.

(b) **Bonus (5 points):** Describe how to achieve the same in the case that any t out of n sons should be able to decrypt. (Note that $\mathbb{G}$ is of prime order q, and thus $\mathbb{Z}_q$ is a field.)

Problem 3: Semi-honest two-party computation from FHE (20 points). Let $(K, E, D, Eval)$ be a fully-homomorphic encryption scheme that is ϵ-semantically-secure for n bit encryptions; namely, $(c_i \leftarrow E_{sk}(x_i))_{i\in[n]} \approx_{c,\epsilon} (c_i \leftarrow E_{sk}(x'_i))_{i\in[n]}$. (The $Eval$ algorithm can take any function $f : \{0,1\}^n \to \{0,1\}^*$ and n bit encryptions $(c_i \leftarrow E_{sk}(x_i))_{i\in[n]}$, and homomorphically evaluate f on the input $(x_1, \ldots, x_n)$.) We say that the scheme is also ϵ-function-hiding, if $Eval$ is probabilistic, and for any sk sampled by K, any $x = (x_1, \ldots, x_n) \in \{0,1\}^n$, encryptions $(c_i \leftarrow E_{sk}(x_i))_{i\in[n]}$, and two functions $f_1, f_2 : \{0,1\}^n \to \{0,1\}$ such that $f_1(x) = f_2(x)$, it holds that $Eval(f_1, (c_i)_{i\in[n]}) \approx_{c,\epsilon} Eval(f_2, (c_i)_{i\in[n]})$.

Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be a 2-party function. Describe a protocol between two parties, Alice and Bob, which hold $x \in \{0,1\}^n$ and $y \in \{0,1\}^n$, respectively, for computing $f(x,y)$. The protocol should protect the privacy of each party's private input, assuming that both parties follow the protocol. Specifically, we require that for any y used by Bob, and any two x, x' such that $f(x,y) = f(x',y)$, Bob can't ϵ-distinguish the two cases. Symmetrically, for any x used by Alice, and any two y, y' such that $f(x,y) = f(x,y')$, Alice can't ϵ-distinguish the two cases.

Problem 4: Computationally unbounded parties, and OT.

(a) (10 points) Consider the following function $F_1(x,y)$, whose tables appears below. The two variables x and y range over $\{0,1,2,3\}$.

Benny (holding x) and Nir (holding y) are both honest, computationally unbounded, and privacy conscious. They wish to engage in a communication protocol such that at the end, both will know $F_1(x,y)$ and nothing else that does not follow from each party's input and the outcome $F_1(x,y)$. Use the characterization given in class to show that cannot be privately computed for these two honest but curious participants.

(b) (10 points) In the same setting as above, Benny and Nir now wish to privately compute the following function, $F_2(x,y)$, whose table appears below. The variable x ranges over

$x \setminus y$	0	1	2	3
0	0	0	1	5
1	2	4	1	6
2	2	3	3	7
3	11	10	9	8

Table 1: $F_1(x, y)$

a subset of $\{0, 1, 2\} \times \{0, 1, 2\}$, while y ranges over $\{0, 1\}$.

$x \setminus y$	0	1
(0,1)	0	1
(0,2)	0	2
(1,0)	1	0
(1,2)	1	2
(2,0)	2	0
(2,1)	2	1

Table 2: $F_2(x, y)$

Show that $F_2(x, y)$ can be privately computed by these honest but curious parties (supply a protocol, no proof required).

(c) (10 points) The function $F_2(x, y)$ are closely related to 1-out-of-2 oblivious transfer. The same Benny and Nir argued in class (without proof) that oblivious transfer is not privately computable if the parties are computationally unbounded. How is that claim compatible with the result of the last section?