CS 355: Topics in Cryptography **Spring 2024**

Problem Set 3

Due: Friday, 10 May 2024 6PM PST (submit via Gradescope)

Instructions: You **must** typeset your solution in LaTeX using the provided template:

`https://crypto.stanford.edu/cs355/24sp/homework.tex`

Submission Instructions: You must submit your problem set via Gradescope. Please use course code **RKN4PX** to sign up. Note that Gradescope requires that the solution to each problem starts on a **new page**.

Bugs: We make mistakes! If it looks like there might be a mistake in the statement of a problem, please ask a clarifying question on Ed.

Problem 1: Somewhat-homomorphic encryption from pairings [10 points]. In this problem, you will construct a "somewhat homomorphic" public-key encryption scheme: it allows computing any number of additions and a single multiplication. Let $\mathbb{G}_1$ be a cyclic group of prime order p and $g \in \mathbb{G}_1$ be a generator of the group. Consider the following two algorithms:

$\mathsf{Gen}(g) \to (\mathsf{pk}, \mathsf{sk})$: Choose random $a, b, c \xleftarrow{R} \mathbb{Z}_p$ such that $c \neq ab \pmod{p}$. Set $g_a = g^a$, $g_b = g^b$, and $g_c = g^c$. Output the public key $\mathsf{pk} = (g, g_a, g_b, g_c)$ and the secret key $\mathsf{sk} = (a, b, c)$.

$\mathsf{Enc}(\mathsf{pk} = (g, g_a, g_b, g_c), m) \to \mathsf{ct}$: Given a message $m \in \mathbb{Z}_p$, choose $r \xleftarrow{R} \mathbb{Z}_p$ and output $\mathsf{ct} = (g^m g_a^r, g_b^m g_c^r)$.

(a) Give a Dec algorithm that takes a secret key sk and a ciphertext $\mathsf{ct} = (u, v)$ and outputs m. Your algorithm needs to be efficient only if the message m lies in some known small space (say $0 \le m < B$ as an integer, for some bound $B = O(\text{polylog}(p))$).

(b) Give an algorithm $\mathsf{Add}(\mathsf{pk}, \mathsf{ct}, \mathsf{ct}') \to \mathsf{ct}_{\text{sum}}$ that takes as input two ciphertexts ct and ct', that are encryptions of $m, m' \in \mathbb{Z}_p$ respectively, and outputs an encryption of $m + m' \bmod p$.

Now let $\mathbb{G}_2, \mathbb{G}_T$ be two other cyclic groups of order p (i.e., $|\mathbb{G}_1| = |\mathbb{G}_2| = |\mathbb{G}_T|$), $e\colon \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ be a pairing, and $h \in \mathbb{G}_2$ and $e(g, h) \in \mathbb{G}_T$ be generators of $\mathbb{G}_2$ and $\mathbb{G}_T$ respectively. Furthermore, let $(\mathsf{pk}', \mathsf{sk}') \leftarrow \mathsf{Gen}(h)$ be the public and secret keys obtained by running Gen using the group $\mathbb{G}_2$. Consider now the following algorithm:

$\mathsf{Mult}(\mathsf{ct}, \mathsf{ct}')$: On input two ciphertexts $\mathsf{ct} = (u, v) \leftarrow \mathsf{Enc}(\mathsf{pk}, m)$ and $\mathsf{ct}' = (u', v') \leftarrow \mathsf{Enc}(\mathsf{pk}', m')$, output the tuple $(w_1, w_2, w_3, w_4) \in \mathbb{G}_T^4$ where

$$w_1 = e(u, u'), \quad w_2 = e(u, v'), \quad w_3 = e(v, u'), \quad w_4 = e(v, v').$$

(c) Let $\alpha_1, \ldots, \alpha_4 \in \mathbb{Z}_p$ such that $w_i = e(g, h)^{\alpha_i}$ (i.e., α_i is the discrete log of w_i in $\mathbb{G}_T$). Show that $m \cdot m' \bmod p$ can be expressed as a *linear function* $\sum_{i=1}^4 C_i \alpha_i$, where the coefficients C_i are independent of m, m'. (You *need not* give an explicit formula for the coefficients C_i.)

(d) Show how to efficiently recover $m \cdot m' \bmod p$ from $w_1, \ldots, w_4$ and the two secret keys sk and sk'. As in Part (a), you can assume that the messages m, m' lie in some known small space.

(e) **Extra credit [3 points].** Show that if the DDH assumption holds in $\mathbb{G}_1$ then $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is a semantically secure public-key encryption scheme.

Problem 2: Conceptual Questions [7 points]. For each of the following statements, say whether it is TRUE or FALSE. Write *at most one sentence* to justify your answer.

(a) Let $\langle P, V\rangle$ be an interactive proof system for a language $\mathcal{L}$ with a *randomized* verifier. If $\langle P, V\rangle$ satisfies perfect completeness (i.e., completeness holds with probability 1) and perfect soundness (i.e., soundness holds with probability 1), then there is an interactive proof system for $\mathcal{L}$ with a *deterministic* verifier.

(b) Let $\langle P, V\rangle$ be a zero-knowledge interactive protocol for some language. The protocol has perfect completeness and soundness error $1/3$. Which of the following are true:

i A malicious verifier interacting with an honest prover will always accept a true statement.

ii An honest verifier interacting with a malicious prover will "learn nothing" besides the statements validity.

(c) If an interactive proof $\langle P, V\rangle$ for an NP language $\mathcal{L}$ is a proof of knowledge with negligible knowledge error, then $\langle P, V\rangle$ has negligible soundness error (i.e., a malicious prover can convince an honest verifier of a false statement with at most negligible probability).

(d) If an interactive proof system has soundness error greater than $1/3$, then it cannot be a proof of knowledge with knowledge error less than $1/3$.

(e) Consider a modified version of Schnorr's signature in which the signing nonce r is computed as $r \leftarrow H(m)$, where $H: \{0,1\}^* \to \mathbb{Z}_q$ is a hash function, m is the message to be signed, and q is the order of the group used for the signature scheme. This deterministic version of Schnorr's signature scheme is secure.

(f) The Fiat-Shamir heuristic (as discussed in class) is a way to construct non-interactive zero-knowledge proofs *without* needing to rely on random oracles.

(g) Consider a hash function $H: \mathcal{W} \to \mathcal{X}$, and the NP-relation $\mathcal{R}_n$ for knowledge of n pre-images of H. Formally, $\mathcal{R}_n$ has instance space $\mathcal{X}^n$, witness space $\mathcal{W}^n$, and is defined by $\{(x \in \mathcal{X}^n, w \in \mathcal{W}^n) : H(w_1) = x_1 \wedge H(w_2) = x_2 \wedge \cdots \wedge H(w_n) = x_n\}$. A SNARG for $\mathcal{R}_n$ must have $o(n)$ verification time.

Problem 3: Understanding Interactive Proofs [15 points]. *(Problems from "The Foundations of Cryptography - Volume 1, Basic Techniques" by Oded Goldreich)*

(a) *The role of verifier randomness:* Let L be a language with a sound and complete interactive proof system where the verifier V is deterministic. Show that $L \in \mathsf{NP}$.

(b) *The role of prover randomness:* Let L be a language with a sound and complete interactive proof system. Show that there exists a sound and complete interactive proof system for L for which the prover P is deterministic.
[**Hint:** Use the fact that P is unbounded.]

(c) *The role of errors:* Let L be a language with a perfectly sound and complete interactive proof system, that is if $x \notin L$, the verifier *never* accepts (not even with negligible probability). Show that $L \in \mathsf{NP}$.

Problem 4: Sigma Protocol for Circuit Satisfiability [10 points]. Let circuit-SAT be the language of satisfiable Boolean circuits[1] :

$$\mathsf{circuit\text{-}SAT} = \left\{C \colon \{0,1\}^n \to \{0,1\} \mid n \in \mathbb{N}, \exists (x_1, \dots, x_n) \in \{0,1\}^n \text{ such that } C(x_1, \dots, x_n) = 1\right\}.$$

Let $\mathsf{Commit} \colon \{0,1\} \times \mathcal{R} \to \mathcal{C}$ be a perfectly-binding and computationally-hiding commitment scheme with message space $\{0,1\}$, randomness space $\mathcal{R}$, and commitment space $\mathcal{C}$. Suppose that there exist Sigma protocols $\langle P_{\mathrm{XOR}}, V_{\mathrm{XOR}} \rangle$ and $\langle P_{\mathrm{AND}}, V_{\mathrm{AND}} \rangle$ for languages $\mathcal{L}_{\mathrm{XOR}}$ and $\mathcal{L}_{\mathrm{AND}}$, respectively, where:

$$\mathcal{L}_{\mathrm{XOR}} = \left\{ (c_1, c_2, c_3) \in \mathcal{C}^3 \;\middle|\; \begin{array}{c} \exists (m_1, m_2, m_3) \in \{0,1\}^3, (r_1, r_2, r_3) \in \mathcal{R}^3 \text{ such that} \\ \forall i \in \{1,2,3\} \;\; c_i = \mathsf{Commit}(m_i; r_i) \text{ and } m_1 \oplus m_2 = m_3 \end{array} \right\}$$

$$\mathcal{L}_{\mathrm{AND}} = \left\{ (c_1, c_2, c_3) \in \mathcal{C}^3 \;\middle|\; \begin{array}{c} \exists (m_1, m_2, m_3) \in \{0,1\}^3, (r_1, r_2, r_3) \in \mathcal{R}^3 \text{ such that} \\ \forall i \in \{1,2,3\} \;\; c_i = \mathsf{Commit}(m_i; r_i) \text{ and } m_1 \wedge m_2 = m_3 \end{array} \right\}.$$

Give a Sigma protocol for circuit-SAT. In addition to describing a protocol, you will also need to show that your protocol satisfies completeness, soundness, and honest-verifier zero-knowledge. [**Hint:** When showing that your protocol is honest-verifier zero-knowledge, you may want to use a hybrid argument. One of your hybrids might rely on the commitment scheme being computationally hiding, and the other hybrid might rely on the underlying Sigma protocols being honest-verifier zero-knowledge.]

Problem 5: Time Spent [1 point for answering]. How long did you spend on this problem set? This is for calibration purposes, and the response you provide will not affect your score.

Optional Feedback [0 points]. Please answer the following questions to help us design future problem sets. You do not need to answer these questions, and if you would prefer to answer anonymously, please use this form. However, we do encourage you to provide us feedback on how to improve the course experience.

(a) What was your favorite problem on this problem set? Why?

(b) What was your least favorite problem on this problem set? Why?

(c) Do you have any other feedback for this problem set?

(d) Do you have any other feedback on the course so far?

[1] You can assume without loss of generality that a Boolean circuit consists of only XOR and AND gates.