Fundamentals of Cryptography: Final

Wednesday Jan 8, 2-4PM

Problem 1 (1pt) Which of the following algorithm can be *stateless and deterministic.*

(a) Encryption algorithm Enc, in a CPA-secure PKE scheme;
(b) Signing algorithm Sign, in a strongly unforgeable signature scheme.

Problem 2 (4pt) State, to the best of your knowledge, the relations between the following cryptographic assumptions. Draw an arrow from assumption A to assumption B if assumption A implies assumption B. Note that the relation is transitive, so if you draw an arrow from A to B and an arrow from B to C, there is no need to draw a third arrow from A to C.

- The existence of OWFs.
- The existence of OWPs.
- The existence of constant-round key exchange protocols.
- The existence of (CPA-secure) public-key encryption schemes.
- The existence of digital signature schemes.

Problem 3 (2pt) Write down the construction of your favorite CPA-secure public-key encryption scheme and the name of the computational assumption it depends on.

Problem 4 (3pt) Garbled Circuits You should state how to garble a boolean circuit. The solution is not unique.

Given the circuit C, for each wire $i \in [n]$, the garbling algorithm generates two random $L_{i,0}, L_{i,1}$ as follows: ____fill the blank____. Output $L_{1,0}, L_{1,1}, \ldots, L_{n_{\text{in}},0}, L_{n_{\text{in}},1}$ as the input labels. And output the garbled circuit $\tilde{C}$ as follows

- For each $i \in \{n_{\text{in}}+1, \ldots, n\}$, generate and output a table as follows: ____fill the blank____. Say the gate function is $g : \{0,1\} \times \{0,1\} \to \{0,1\}$, and the gate takes wires j_1, j_2 as inputs.
- For each output wire $i \in \{n - n_{\text{out}} + 1, \ldots, n\}$, output ____fill the blank____.

Formalization of a circuit (for problem 4). A circuit has n wires, including n_{in} input wires, n_{out} output wires and $n - n_{\text{in}} - n_{\text{out}}$ intermediate wires. W.l.o.g., the wires are indexed by $1, \ldots, n$. Given the input x, the value of the i-th wire, denoted by v_i, and the output of the circuit are determined as follows. For a boolean circuit, the input is in $\{0,1\}^{n_{\text{in}}}$. For an arithmetic circuit over $\mathcal{R}$, the input is in $\mathcal{R} \in \{0,1\}^{n_{\text{in}}}$.

- For each $i \leq n_{\text{in}}$, the i-th wire is the i-th input wire, so $v_i = x_i$.
- For each $i > n_{\text{in}}$, the i-th wire is the output of a gate. Say the gate function is g, and the gate takes wires $j_1, \ldots, j_t$ as inputs ($j_1 \leq \cdots \leq j_t < i$). Then $v_i = g(v_{j_1}, \ldots, v_{j_t})$.
- The output of the circuit is $(v_{n-n_{\text{out}}+1}, \ldots, v_n)$.

Problem 5A and 5B are mutually exclusive. Solve one of them.

Problem 5A (5pt) Candidate Symmetric-key Construction Inspired by Lattice Here is a candidate (keyed) OWF construction. The key of the construction consists of $2n$ random $\lambda \times \lambda$ invertible matrixes $k = (M_{i,b})_{i\in[n],b\in\{0,1\}}$ in a given field $\mathbb{F}$. For each input $x \in \{0,1\}^n$, $F_k(x) = M_{1,x_1} M_{2,x_2} \dots M_{n,x_n}$.

Part A. Let $n = \lambda$. Let $\mathbb{F} = \mathbb{Z}_p$ for a prime $p = \text{poly}(\lambda)$. Prove that F is not a PRF.

Part B. Set $n, \lambda, \mathbb{F}$ as in part A. Define keyed function H_k such that $H_k(x)$ is the first (top left) entry of $F_k(x)$. Is H a PRF?

Problem 5B (5pt) Show that one of the following two implies the other.

- Existence of CPA-secure public-key encryption schemes
- Existence of semi-honest 2-message 1-out-of-2 oblivious transfer protocols

Problem 6 (5pt) Positional Secret Sharing (PoSS) PoSS is a highly specialized secret sharing problem.

- Let n denote the number of parties, let k denote the threshold, let ℓ denote secret length.
- The distribution algorithm takes as inputs the parameter (n, k, ℓ), the index of a special party $i^* \in [n]$ and k secret messages $m_1, \dots, m_k \in \{0,1\}^\ell$, outputs n shares $s_1, \dots, s_n$.
- For any subset $T = \{i_1, \dots, i_k\} \subseteq [n]$ of k parties, where $1 \le i_1 < i_2 < \cdots < i_k \le n$:

  (Correctness) If $i^* = i_\alpha \in T$, the α-th secret m_α can be recovered from $(s_{i_1}, \dots, s_{i_k})$.

  (Privacy) Nothing else about $(m_1, \dots, m_k)$ can be recovered from $(s_{i_1}, \dots, s_{i_k})$.

Construct a perfectly secure PoSS in the plain model. Explicitly state the cryptographic assumptions you used, if any. Explicitly state the size of s_i (should be at most $\text{poly}(n, k, \ell)$).

Problem 7 (5pt) 1-out-of-2^n Oblivious Transfer Let Π be a secure 2-message 1-out-of-2 oblivious transfer protocol. For simplicity, we focus on semi-honest security in this problem.

Part A. Construct a 1-out-of-2^n oblivious transfer protocol based on Π. Your protocol should make only black-box use of Π, and should not rely on any other assumption.

Part B. Say Π is a "rate-1" protocol. Its communication cost is highly optimized. If the two messages are ℓ-bit long, the total communication complexity of Π is $\ell + \text{poly}(\lambda)$.

Construct a 1-out-of-2^n oblivious transfer protocol based on Π. Your protocol should make only black-box use of Π, and should not rely on any other assumption. When all the 2^n messages are ℓ-bit long, the total communication complexity of your protocol should be at most $\ell + \text{poly}(n, \lambda)$.

Problem 8 (5pt) MPC in the Head Consider the following zero-knowledge proof protocol, in which the prover emulates a MPC protocol in his head.

- Let ϕ denote the statement and let w denote the witness (i.e. $\phi(w) = 1$). The verifier knows ϕ and the prover is given both ϕ and w.
- The prover computes additive shares $w_1, \ldots, w_n$ of the witness w. He locally emulating a n-party MPC protocol Π that computes the function $(w_1, \ldots, w_n) \mapsto \phi(w_1 + \cdots + w_n)$. Let $V_1, \ldots, V_n$ be the views of the n parties in the emulation.
- The prover and the verifier use a $\lfloor \frac{n-1}{2} \rfloor$-out-of-$n$ OT protocol. The prover acts as the sender and picks $V_1, \ldots, V_n$ as his messages. The verifier chooses a random size-$\lfloor \frac{n-1}{2} \rfloor$ subset $T \subseteq [n]$. The verifier learns V_i for each $i \in T$.
- The verifier accepts if and only if the views $(V_i)_{i \in T}$ are consistent, and every party in the opened views outputs 1.

Part A. Prove that the above is a zero-knowledge proof protocol.

For simplicity, assume the underlying OT is UC-secure. Therefore, it suffices to analyze the protocol in the OT-hybrid model. There is an extra trusted party (so-called ideal functionality), who takes $V_1, \ldots, V_n$ from the prover, takes $T \subseteq [n]$ from the verifier, and sends $(V_i)_{i \in T}$ to the verifier.

You need to specified the following details: What is the minimum requirement of Π? How to set n? How large is the soundness error of your protocol?

Part B. The above protocol is also a proof of knowledge. Describe the extractor.