MIT 6.5620/6.875/18.425: Cryptography and Cryptanalysis **Due: Nov. 20, 2023**

Problem Set 5

Total Number of Points: 100.

Collaboration Policy: Collaboration is allowed and encouraged in small groups of at most three students. You are free to collaborate in discussing answers, but you must write up solutions on your own and must specify in your submission the names of any collaborators. Do not copy any text from your collaborators; the writeup must be entirely your work. Do not write down solutions on a board and copy them verbatim into LaTeX; again, the writeup must be entirely your own words and your own work and should demonstrate a clear understanding of the solution. Additionally, you may make use of published material, provided that you acknowledge all sources used. Of course, scavenging for solutions from prior years is forbidden.

Problem 1. Better NIZK Proofs? (40 points)

One of you made a nice suggestion in class, to use pseudorandom generators to shrink the CRS in NIZK proofs. Let's see if that works! But first let's set up some notation. Let λ denote the security parameter for this problem. An NIZK proof system for an $\mathcal{NP}$ language L is defined by a pair of algorithms $(\mathcal{P}, \mathcal{V})$ where:

- $\mathcal{P}$ is the prover algorithm that is given a uniformly random string $\mathsf{crs} \leftarrow \{0,1\}^\ell$, an NP statement $x \in L \cap \{0,1\}^n$ and its NP witness $w \in \{0,1\}^m$, and produces an NIZK proof π. That is,
$$\pi \leftarrow \mathcal{P}(\mathsf{crs}, x, w)$$
- $\mathcal{V}$ is the verifier algorithm that is given the crs, x and the NIZK proof π, outputs "accept" or "reject". That is,
$$\mathcal{V}(\mathsf{crs}, x, \pi) \in \{\mathsf{ACC}, \mathsf{REJ}\}$$

Such a proof system should satisfy three properties: completeness, soundness and zero knowledge. We restrict ourselves to NIZK proof systems with perfect completeness.

In the Blum-Feldman-Micali NIZK proof system we saw in class, the CRS had length $\ell = n \cdot p(\lambda)$ for some polynomial p. (Convince yourself that this is the case.) One could try to convert any such NIZK proof system into another one, call it $(\mathcal{P}', \mathcal{V}')$, where the CRS is smaller, i.e., has size $\ell' = p'(\lambda)$ for some polynomial p' independent of the instance size n, in the following way. Let $G : \{0,1\}^{\ell'} \to \{0,1\}^\ell$ be a pseudorandom generator.

- $\mathcal{P}'(\sigma, x, w)$ computes $\mathsf{crs}' = G(\sigma)$ and runs $\mathcal{P}(\mathsf{crs}', x, w)$.
- $\mathcal{V}'(\sigma, x, \pi)$ computes $\mathsf{crs}' = G(\sigma)$ and runs $\mathcal{V}(\mathsf{crs}', x, \pi)$.

(a) (4 points) Does $(\mathcal{P}', \mathcal{V}')$ always satisfy completeness? Prove your answer.

(b) (18 points) Does $(\mathcal{P}', \mathcal{V}')$ always satisfy soundness? Prove your answer.

(c) (18 points) Does $(\mathcal{P}', \mathcal{V}')$ always satisfy zero knowledge? Prove your answer.

We will assume SAT is not in BPP, and in all cases, to show a "yes" answer, you have to show that for *every* NIZK proof system $(\mathcal{P}, \mathcal{V})$, the compiled proof system $(\mathcal{P}', \mathcal{V}')$ is complete/sound/zero-knowledge. To show a "no" answer, you have to show that there exists *some* NIZK proof system $(\mathcal{P}, \mathcal{V})$ such that the compiled proof system $(\mathcal{P}', \mathcal{V}')$ is *not* complete/sound/zero-knowledge.

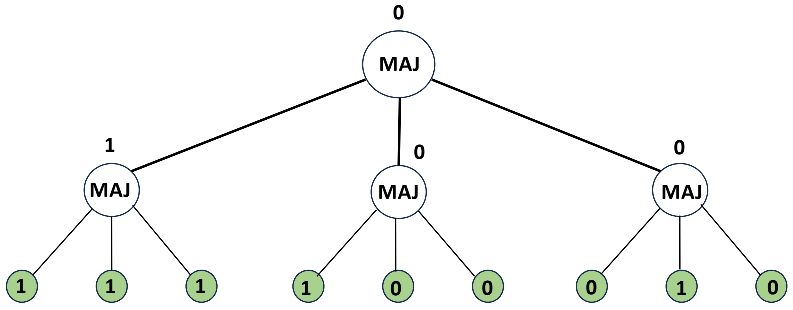

Figure 1: The function f representing a hierarchical access structure on 9 users. A set of users corresponds to a Boolean labeling of the leaves. For example, the labeling of the leaves above corresponds to the set of users $T = \{1, 2, 3, 4, 8\}$. An internal node is labeled 1 if and only if the majority of its children are labeled 1. A given set is in the access structure if and only if the root is labeled 1. (In this case, since the root has label 0, T is not in the access structure.)

Problem 2. Secret Sharing, Generalized (30 points)

In class, we saw the notion of threshold secret sharing. Given n users (numbered 1 through n), define the (t, n)-threshold access structure to be the collection of sets

$$\mathcal{A}_{\mathsf{threshold}} = \{S \subseteq [n] : |S| \geq t\}$$

The property of a secret sharing scheme, then, is that when a set $T \subseteq [n]$ of users come together, they can reconstruct the secret from their shares if and only if $T \in \mathcal{A}_{\mathsf{threshold}}$.

In this problem, we will generalize secret sharing to other access structures. In general, an access structure $\mathcal{A}$ is a monotone collection of subsets of $[n]$. Here, monotonicity refers to the condition that if $\mathcal{A}$ contains a set T, it also contains all supersets of T. (Do you see why this has to be the case?)

- We will associate to a subset $T \subseteq [n]$ its characteristic vector $x_T \in \{0, 1\}^n$. For example, letting $n = 4$ and $T = \{1, 3\}$, we have $x_T = (1010)$.
- We will associate to an access structure $\mathcal{A}$ a (monotone) Boolean function $f_{\mathcal{A}} : \{0, 1\}^n \to \{0, 1\}$ where

  $$T \in \mathcal{A} \text{ if and only if } f_{\mathcal{A}}(x_T) = 1 .$$

  For example, in the case of the threshold access structure, $f_{\mathcal{A}}$ is simply the threshold function which outputs 1 if and only if the input has at least t ones.

(a) (15 points) Let $n = 3^m$ be the number of parties, and let the hierarchical access structure be defined by the majority-of-majorities function (see Figure 1 for an example with $m = 2$ and $n = 9$). Construct a secret sharing scheme for the hierarchical access structure on $n = 3^m$ users for arbitrary depth m. The shares in your scheme should have bit-length polynomial in n.

(b) (15 points) Let the conjunctive normal form (CNF) access structure be defined by a monotone CNF formula Ψ with n variables and m clauses (monotone in the sense that the formula has no negations). Construct a secret sharing scheme for the CNF access structure. The shares in your scheme should have bit-length polynomial in n and m.

Problem 3. Oblivious Transfer Pro Max (30 points)

Recall that in class, we learned about 1-out-of-2 Oblivious Transfer (OT). In an OT protocol, a sender has two message bits $m_0, m_1 \in \{0,1\}$, and a receiver has a choice bit $b \in \{0,1\}$. The sender wants to send m_b to the receiver while satisfying correctness (the receiver obtains m_b), sender's privacy (the receiver gains no knowledge about the message m_{1-b}), and receiver's privacy (the sender gains no knowledge about the choice bit b).

In this problem, we focus on achieving security against *honest-but-curious* senders and receivers.

(a) (10 points) Show how you can use any 1-bit OT scheme to build an ℓ-bit OT scheme for transferring ℓ-bit messages $m_0, m_1 \in \{0,1\}^\ell$. Here $\ell = \ell(\lambda)$ is a (possibly large) polynomial in security parameter λ. Your scheme can only invoke the given 1-bit OT scheme at most $\lambda \ll \ell$ times. You can assume the existence of a pseudorandom generator.

(b) (20 points) A 1-out-of-n secret sharing scheme is one where the sender has n messages $m_0, \ldots, m_{n-1} \in \{0,1\}^\ell$ and the receiver wants the i^{th} message m_i.

You are given a 1-out-of-2 OT scheme with ℓ-bit messages. Show how to construct a 1-out-of-n OT scheme for any integer $n \geq 2$. You can assume the existence of a PRF family. For full credit, your scheme must invoke the 1-out-of-2 OT scheme at most $O(\log n)$ many times.