CS 409M : Introduction to Cryptography Fall 2024

Assignment 7
Testing Soundness of your Knowledge

Instructor: Sruthi Sekar *TAs:* Giriraj Singh, Gyara Pragathi, Nilabha Saha

Problem 1: Malicious-Verifier ZKP for QR

Consider the language of quadratic residues over $\mathbb{Z}_N^*$, defined as follows:

> Let N be an integer such that $N = pq$ for two distinct primes p and q, and let $x \in \mathbb{Z}_N^*$ be an element. Define the language
>
> $$\mathcal{L}_{\mathsf{QR}} := \{x \in \mathbb{Z}_N^* : \exists\ y \in \mathbb{Z}_N^* \text{ such that } y^2 \equiv x \pmod{N}\}$$

Provide a Zero-Knowledge Proof (ZKP) system for the language $\mathcal{L}_{\mathsf{QR}}$, assuming the presence of a potentially malicious verifier.
Hint: Think along the same lines as the ZKP for Graph isomorphism discussed in class.

Problem 2: HVZKP for QNR

Consider the language of quadratic non-residues over $\mathbb{Z}_N^*$, defined as follows:

> Let N be an integer such that $N = pq$ for two distinct primes p and q, and let $x \in \mathbb{Z}_N^*$ be an element. Define the language
>
> $$\mathcal{L}_{\mathsf{QNR}} := \{x \in \mathbb{Z}_N^* : \nexists\ y \in \mathbb{Z}_N^* \text{ such that } y^2 \equiv x \pmod{N}\}$$

Design an Honest Verifier Zero-Knowledge Proof (HVZKP) protocol for the language $\mathcal{L}_{\mathsf{QNR}}$.
Hint: Think along the same lines as the ZKP for Graph non-isomorphism discussed in class.

Problem 3: Bad Randomness Attack on Schnorr Signatures

Let (sk, pk) be a key pair for the Schnorr signature scheme, defined as below (from lecture 17).

- $\mathsf{Gen}(1^n)$: Generate $\mathcal{G} \to (\mathbb{G}, q, g)$, Pick random oracle $H : \{0,1\}^* \to \mathbb{Z}_q$ implicitly. Generate $x \in_R \mathbb{Z}_q$ and compute $y = g^x$. Output $pk = (\mathbb{G}, q, g, y)$, and $sk = (\mathbb{G}, q, g, x)$.
- $\mathsf{Sign}(sk, m)$: Choose $k \in_R \mathbb{Z}_q$, compute $I = g^k$, $r = H(I, m)$, $s := (rx + k) mod q$ and output $\sigma = (r, s)$.
- $\mathsf{Vrfy}(pk, m, \sigma)$: output 1 if and only if $H(g^s \cdot y^{-r}, m) = r$.

Suppose the signing algorithm is faulty and chooses dependent values for k_t in consecutively issued signatures. In particular, when signing a message m_0 the signing algorithm chooses a uniformly random k_0 in $\mathbb{Z}_q$, as required. However, when signing m_1 it choose k_1 as $k_1 \leftarrow a \cdot k_0 + b$ for some known $a, b \in \mathbb{Z}_q$. Show that if the adversary obtains the corresponding Schnorr message-signature pairs (m_0, σ_0) and (m_1, σ_1) and knows a, b and pk, it can learn the secret signing key sk, with high probability.
This attack illustrates why it is important to derandomize signature schemes derived from Sigma protocols, in practice. It ensures that the signer is not dependent on the security of its entropy source (i.e., the distribution used to pick k).

Problem 4: Unique responses of Sigma Protocols

Let Π be a Sigma protocol. We say that Π has *unique responses* if for every statement $x \in L$ and for every pair of accepting conversations (com, c, r) and (com, c, r') for x, we must have $r = r'$.

1. Prove that Schnorr's Sigma protocol, as we described in class, has unique responses.
2. Prove that the Chaum-Pedersen protocol, as we described in class, has unique responses.

Problem 5: Commitment or not?

Consider the following (non-interactive) commitment schemes $(\mathsf{Commit}, \mathsf{Receive})$ for single-bit messages (as defined in the class). Which of the following schemes are computational hiding and perfect binding commitment schemes? Explain. Further, if not perfectly binding, are these schemes computationally binding (defined as below)?

Comutational Binding: A commitment scheme $(\mathsf{Commit}, \mathsf{Receive})$ is said to be computationally binding if for all PPT adversary A, there exists a negligible function negl such that $\Pr[(m_0, m_1, r_0, r_1) \leftarrow A(1^n) : (m_0 \neq m_1) \wedge \mathsf{Commit}(m_0, r_0) = \mathsf{com} = \mathsf{Commit}(m_1, r_1)] \leq \mathsf{negl}(n)$

1. Given a one-way function $f : \{0,1\}^{n+1} \to \{0,1\}^{n+1}$, $\mathsf{Commit}(b, r) := f(b||r)$, for bit b and randomness $r \in \{0,1\}^n$.
2. Given a random oracle $H : \{0,1\}^{n+1} \to \{0,1\}^n$, $\mathsf{Commit}(b, r) := H(b||r)$, for bit b and randomness $r \in \{0,1\}^n$.
3. Given a one-way permutation $f : \{0,1\}^n \to \{0,1\}^{n+1}$, $\mathsf{Commit}(b, r) := (b \oplus r_1 \oplus r_2 \oplus \cdots \oplus r_n, f(r))$, for bit b and randomness $r = r_1||r_2||\ldots||r_n \in \{0,1\}^n$.

Problem 6: ZKP for Knight's Move Sudoku

Notation: The notation $[n]$ indicates all natural numbers from 1 to n (both inclusive), that is, $[n] = \{x \in \mathbb{N} | 1 \leq x \leq n\}$.

Knight's Move Sudoku is a variant on the popular logic puzzle Sudoku. For a Knight's Move Sudoku puzzle for positive integer n, you are given as input an $n^2 \times n^2$ grid of cells, where the cells are partitioned into n^2 disjoint $n \times n$ subgrids. Some of the cells begin filled with integers from 1 to n^2.

A solution is a way to assign an integer from 1 to n^2 to each cell which hasn't already been filled in a way that:

1. Each row contains each of the integers from 1 to n^2 exactly once.
2. Each column contains each of the integers from 1 to n^2 exactly once.
3. Each of the n^2 different $n \times n$ subgrids contains each of the integers from 1 to n^2 exactly once.
4. For all cells that are a "knight's move" away from each other, the entries are distinct. Two cells $(i_1, j_1) \in [n^2] \times [n^2]$, $(i_2, j_2) \in [n^2] \times [n^2]$ are a "knight's move" away if either (a) $|i_1 - i_2| = 1$ and $|j_1 - j_2| = 2$ or (b) $|i_1 - i_2| = 2$ and $|j_1 - j_2| = 1$.

You can find an example demonstrating a knight's move at https://masteringsudoku.com/chess-sudoku/.

Design a zero-knowledge protocol for Knight's Move Sudoku. Your protocol should be computationally zero knowledge, have perfect completeness, and have soundness error $1 - p(n)$ for a non-negligible function p. You must prove that your construction satisfies all of the above properties. You may assume the existence of one-way functions.