Zero Knowledge Proofs

We previously went over FHE, which allows us to store encrypted data and have the server do computations on it without decrypting

- How do we verify that the platform is doing the instructed computation?
- Can we efficiently verify a computation was done correctly?

For a time T computation, we want to verify it in time $<< T$

- Not every T computable language has a proof / witness of size T

For much of history, our proof systems have been a list of formulas that go from axioms to our desired result

- A different proof system is a **zero knowledge proof**
  - It reveals no information beyond the validity of the statement
  - This means that we could have generated it on our own
- We use interactive proofs, where the proof arises from an interaction between two parties rather than
  - We have a prover and a verifier where the prover is trying to convince the verifier the problem instance is true
  - The verifier is trying to verify that the prover is not dishonest / cheating

Interactive Proof System

An interactive proof system for a language L consists of an interactive verifier algorithm ν and an interactive algorithm $\mathcal{P}$ which exchange series of messages $m_1, \ldots, m_k$

- The verifier's computations may also depend upon private random bits not rebealed to the prover

$(P, \eta(r))(x) = 1$ is the event that the verifier η with randomness r accepts the interactive proof

There are two properties that have to hold:

- **Completeness**: for all $x \in L$:

$$\Pr\left[(P, V(r))(x) = 1\right] \geq 2/3$$

- **Soundness**: for all $x \notin L$ and all malicious P^*:

$$\Pr\left[(P^*, V(r))(x) = 1\right] \leq 1/3$$

The numbers $2/3$ and $1/3$ are arbitrary, since by repeating the interactive proof we can use the Chernoff bound to get any probability p

- The class IP is the set of all languages that have an interactive proof
- $NP \subseteq IP$
- In fact, we actually have that $IP = PSPACE$ from Shamir '90

Zero Knowledge

For all malicious and powerful V^*, there exists a ppt algorithm Sim such that for all $x \in L$:

$$\{\text{View}_{V^*}(P(x), V^*)\} \approx \{\text{Sim}(x)\}$$

where the LHS is the view of the malicious verifier in its interaction with the prover

We can give a zero-knowledge proof for a speicfic NP-complete languag called 3Col:

- This language is the set of all graphs such that the vertices can be colored by three colors
- A proof would be a coloring $C : V \to \{1, 2, 3\}$
- A zero knowledge proof would be:

1. Prover chooses a random permutation and for every i, places the color $\pi(C(i))$ in an opaque locked box
2. All locked boxes are sent to the verifier

3. The verifier chooses a random edge $(i, j) \in E$ and sends (i, j) to the prover
4. The prover sends the keys that open only box i and j
5. The verifier accepts iff the colors in these boxes are distinct and legal

The only thing the verifier learns is two distinct random colors, and could have simulated this on its own

- By doing repetitions, we can amplify the soundness

There is a digital analogue to the opaque locked boxes:

Commitment Schemes

A **commitment scheme** corresponds to a message space consisting of a pair of algorithms:

- Gen: a ppt algorithm that takes as input λ and outputs public parameters $pp \in \{0,1\}^N$, where $N \in \text{poly}(\lambda)$
- Com: a polynomial time function that takes as input pp, a message m, a randomness r, and outputs a commitment

The idea is the commitment is sent to the receiver, and later, the receiver could get the key k and the randomness r and open the commitment to receive the message

Two properties have to hold:

- Hiding: For every $m_0, m_1 \in \mathcal{M}$:

$$(pp, Com(pp, m_0, r_0)) \approx (pp, Com(pp, m_1, r_1))$$

- Binding: for every ppt adversary $\mathcal{A}$, the probability is negligible that $\mathcal{A}$ can take pp and generate m_0, r_0, m_1, r_1 such that $m_0 \neq m_1$ and there commitment is the same
    - The idea is that the sender cannot later try to come up with a different r to try to claim their commitment was something else

LWE leads to the following commitment scheme for the message space $\mathcal{M} = \{0, 1\}$:

- Gen chooses a random matrix A in the field $\mathbb{Z}_q^{m \times n}$ and a random vector u from $\mathbb{Z}_q^m$ that form $p$$
- Com is just $As + e + bu$ where b is the message and (s, e) is the randomness