Today: Zero-knowledge interactive proofs

[Goldwasser-Micali-Rackoff 85]

So far:
- One time pad
- OWF + hard core bits
  - Blum-Micali construction from one-way permutations
  - GGM construction
  - PRF
  - HW
  - Semantic secure encryption

Today: Magic of zero-knowledge proofs

What is a proof? A proof for a statement \( x \in L \) consists of a sequence of symbols \( w \) and a polynomial time verifier \( V \) s.t. \( V(x, w) = 1 \), and if \( x \not\in L \) then \( \forall w \ V(x, w) = 0 \).

Proofs are fundamental objects in theory of computation, and is formalized by the complexity class \( \text{NP} \), which consist of all statements that have poly-size proofs:

Def: The complexity class \( \text{NP} \) consists of all languages \( L \) s.t.
\exists \text{const } c \in \mathbb{N} \ \exists \text{Turing machine } V \text{ st. } \forall x \in \{0, 1\}^n

- If \( x \in L \) then \( \exists w \in \{0, 1\}^n \) s.t. \( V(x, w) = 1 \) after running in time at most \( n^c \) (\( n = |x| \))
- If \( x \notin L \) then \( \forall w \in \{0, 1\}^n \) \( V(x, w) = 0 \).

Goal: Construct proofs \( \omega \) that reveal no information.

Impossible! \( \omega \) itself is information; i.e., before receiving \( \omega \), \( V \) could not convince anyone that \( x \in L \) but with \( \omega \) he can. So he learned something:

How to convince that \( x \in L \).

Solution: Change the model!

Interactive Proofs

Allow proofs to be interactive, i.e., the prover receives queries from the verifier and replies to them:

\[
P \xleftarrow{g_1} V \\
\xrightarrow{a_1} \\
\vdots
\]

\[
P \xleftarrow{g_2} \\
\xrightarrow{a_2} \\
\vdots
\]
Importantly: Verifier is allowed to be probabilistic (without adding randomness interactive proofs are the same as non-interactive ones since the prover can compute the verifier's queries on his own).

We allow a probability of error: There is a negligible probability of accepting false statements.

\textbf{Def}: An interactive proof system \((P,V)\) for a language \(L\) consists of interactive parties (Turing machines): \(P\) (the prover) and \(V\) (the verifier). The verifier is restricted to be PPT (no restrictions on \(P\)), and the following properties hold:

- **Completeness**: \(\forall x \in L \quad \Pr[P^* | (P,V)(x) = 1] = 1\)
  
  \(V\) accepts

- **Soundness**: \(\forall x \notin L \quad \Pr[P^* | (P,V)(x) = 1] \leq \frac{1}{2}\)

  possibly cheating

\textbf{Note}: Soundness can be made arbitrarily low by repetition.

\(\text{IP} = \text{all languages that have an interactive proof}\)
Clearly NPCIP.

It turns out that IP is much more powerful:

Thm [Lund-Fortnow-Karloff-Nisan'90, Shamir'90]:

$$\text{IP} = \text{PSPACE}.$$

Today: Interactive proofs are useful for zero-knowledge (ZK).

Intuitively, an interactive proof is ZK if the verifier did not learn anything from the proof, except the validity of the statement.

**Def.** Given an interactive proof $\langle P, V \rangle$ for $L$, for any input $x \in L$ we denote by $\text{View}_{P,V}(x)$ the random variable describing the view of $V$ after interacting with $P$ on input $x$ (i.e. $\text{View}_{P,V}(x)$ consists of the transcript & $V$'s coin tosses, and input $x$).

More generally, $V^*$ (possibly cheating verifier) $V^* \neq x \in L$ we denote by $\text{View}_{P,V^*}(x)$ the random variable describing the view of $V^*$ after interacting with $P$ on input $x$.

Loosely speaking a proof system $\langle P, V \rangle$ for $L$ is ZK
If \( \exists x \in L \) the verifier could generate (simulate) this view on her own, without interacting with \( P \), and therefore did not learn anything from the interaction.

There are many flavors of ZK.

**Def:** An interactive proof \((P,V)\) for \( L \) is said to be

- **perfect**
- **honest-verifier ZK** if \( \exists \) PPT alg \( S \) s.t. \( \forall x \in L \):
  - statistical
  - computational

\[
\text{view}_{(P,V)}(x) \equiv S(x) \uparrow
\]

- identical
- statistically close
- computationally indistinguishable

**Def:** An interactive proof \((P,V)\) for \( L \) is said to be

- **perfect**
- **ZK** if \( \forall \) PPT (non-uniform) \( V^* \) \( \exists \) PPT (non-uniform) alg \( S \) s.t. \( \forall x \in \{0,1\}^n \) \( L \)
  - \( \forall z \in \{0,1\}^{\text{poly}(n)} \)

\[
\text{view}_{(P,V)}(x,z) \equiv S(x,z) \uparrow
\]

- identical
- statistically close
- comp. indist.
Example: Graph Isomorphism

\[ L = \{ (G_1, G_2) : G_1 \cong G_2 \} \]

\[ \exists \text{ bijection } \pi : V(G_1) \rightarrow V(G_2) \]

\[ \text{ such that } (u, v) \in E(G_1) \text{ iff } (\pi(u), \pi(v)) \in E(G_2) \]

Deciding if two graphs are isomorphic is considered to be a hard problem.

A classical proof: \( \pi \) (this is a witness for \((G_1, G_2) \in L\)).

ZK proof

\[ P \quad (G_1, G_2) \in L \quad \forall \]

Sample a random graph \( H \) isomorphic to both \( G_1 \) & \( G_2 \)

\[ H = \pi^*(G_1) \quad \overset{H}{\longrightarrow} \]

random bijection

\[ \overset{b}{\longleftrightarrow} \quad b \leq 2013 \]

If \( b = 0 \) \( \pi'' = \pi^* \quad \overset{\pi''}{\longrightarrow} \)

If \( b = 1 \) \( \pi'' = \pi^* \pi^{-1} \)

Check that \( \pi'' \)

is a witness for homomorphism of \( H \) & \( G_b \)
It is easy to see that this is an interactive proof system for $L$ (i.e., satisfies completeness & soundness).

**Perfect ZK:** $S(G_1, G_2, z)$

- Choose at random $b^* \in \{0, 1\}$ & a random bijection $\pi^*$. Let $H = \pi^*(G_{b^*})$
- Compute $b = V((G_1, G_2, z, H))$
- If $b = b^*$ output $(H, b^*, \pi^*)$
- Else try again

Try at most $\mathcal{N}$ (input length) times.
- If all fail output $\bot$.

**Claim:** $\forall (G_1, G_2) \in L: S(G_1, G_2, z) \equiv \text{View}_{(p, v^*)} (G_1, G_2, z)$ $2^{-\mathcal{N}}$-close

**Pf:** Note that if $S(G_1, G_2, z) \neq \bot$ then the dist. of $S(G_1, G_2, z)$ is identical to $\text{View}_{(p, v^*)} (G_1, G_2, z)$

Moreover, $\Pr[ S(G_1, G_2, z) = \bot] = 2^{-\mathcal{N}}$

**Remark:** Can make it perfect ZK if $S$ is expected poly-time.
**Remark 2:** $S$ is a universal simulator, uses $V$ in a black-box.

**Remark 3:** Soundness is only $1/2$. Can reduce soundness to $2^{-n}$ via $n$ repetitions.

$\mathbb{Z}_k$ is preserved under sequential repetition (but not under parallel repetition).

The reason it is preserved under sequential repetition is the auxiliary input.

**Example:** Quadratic residues mod $N$ ($QR_N$)

$$L = \{(y, N) : \exists x \text{ s.t. } y = x^2 \mod N\}$$

Deciding if $(y, N) \in L$ for $N = p \cdot q$ (p, q random large primes) & $y$ random element in $\mathbb{Z}_N^*$ (with Jacobi symbol $+1$) is believed to be hard (assuming Factoring is hard).

The first public key encryption scheme proposed by Goldwasser & Micali (1982) relies on the hardness of $QR_N$. 
A classical proof for \((y, N) \in L: x \text{ s.t. } y = x^2 \mod N\):

**ZK proof:**
\[ P \quad (y, N) \in L \quad V \]

Choose at random \( r \leftarrow \mathbb{Z}_N^* \)
\[ \frac{y = r^2 \mod N}{b} \quad b \leftarrow \{0, 1\} \]
\[ \frac{r \cdot x^b}{V} \]

Accept iff
\[ y^b = (r \cdot x^b)^2 \]

It is easy to see that this is an interactive proof (i.e., satisfies completeness & soundness).

**Perfect ZK:** \( S((y, N), z) \):

Choose at random \( b^* \leftarrow \{0, 1\} \)
Choose at random \( r \leftarrow \mathbb{Z}_N^* \), and let
\[ V = r^2 y^{b^*} \mod N \]

Compute \( b = V^*((y, N), z, V) \).
If \( b = b^* \) output \((V, b^*, r)\)
else, try again.

Repeat at most \( n \) times.
As w. graph isomorphism, it is $2^{-n}$-statistical ZK or perfect ZK w.r.t. expected PPT simulator.

Next time: Every proof can be converted into a (comp.) ZK interactive proof assuming OWF!