Lec 12 Zero-Knowledge Proof (ZKP)

NP-language L: $\exists$ poly-time M $\forall x \in \{0,1\}^n$

$x \in L \Longleftrightarrow \exists\, w \in \{0,1\}^{poly(n)}$. $M(x, w) \to 1$

witness

e.g. 1 QR

$(G = \mathbb{Z}_{pq}^*, g) \in QR$

iff $\exists a$ $g = a^2$ in $\mathbb{Z}_{pq}^*$

e.g. 2 Grape isomorphism

G, G' are isomorphism

iff $\exists$ permutation π s.t. $\pi \circ G = G'$

$(u,v) \in G \Leftrightarrow \pi(u), \pi(v) \in G'$

(Non-interactive) Honest-Verifier Perfect Zero-Knowledge Proof

Malicious-Verifier Statistical Proof of Knowledge

Computational Argument

Argument of Knowledge

Argument:

soundness holds only against p.p.t. prover

"Classical" Proof

x

Prover   Verifier

$\xrightarrow{\text{witness } w}$

check $M(x, w) = 1$

unbounded   poly-time

1) $\forall x \in L \quad \exists P \quad (P(x), V(x)) \to 1$

2) $\forall x \notin L \quad \forall P \quad (P(x), V(x)) \to 0$

NP-language

Interactive Proof

x

Prover   Verifier

unbounded   poly-time

1) $\forall x \in L \quad \exists P \quad \Pr[(P(x), V(x)) \to 1] \geq 2/3$

2) $\forall x \notin L \quad \forall P \quad \Pr[(P(x), V(x)) \to 0] \geq 2/3$

IP - language

$=$

PSPACE

Multi-prover IP

Prover 1   Verifier

Prover 2

$MIP = NEXP$

QR

Q0 $h/g \in QR$

Q1 $h \in QR$

Simulator S

g, a Prover $\qquad$ Verifier g

$$\text{Prover} \xrightarrow{h = g \cdot r^2} \text{Verifier}$$

$$\text{Prover} \xleftarrow{b \in \{0,1\}} \text{Verifier}$$

$$\text{Prover} \longrightarrow \text{Verifier}$$

$$r \cdot a^b = \begin{cases} r & \text{if } b=0 \\ r \cdot a & \text{if } b=1 \end{cases}$$

$\exists$ p.p.t. Simulator S s.t. $\forall x \in L$

$$\mathrm{View}_V(P(x,w), V(x)) \approx S(x)$$

1) Perfect ZK $\equiv$
2) Statistical ZK $\approx_s$
3) Computational ZK $\approx_c$

Efficiency $P(g, \text{witness})$ runs in poly-time

Completeness if $g \in QR$ $\quad (P(g,a), V(g)) \to 1$

Soundness if $g \notin QR$, $\forall$ unbounded P^* $\quad \Pr[(P^*(g), V(g)) \to 1] \leq \frac{1}{2}$

Zero-Knowledge Verifier's View can be <u>Simulated</u> without interacting with the prover

Graph Isomorphism

G, G', π — Prover

G, G' — Verifier

pick τ

$G'' = \tau \circ G$

Prover $\xrightarrow{G''}$ Verifier

Verifier $\xrightarrow{b \in \{0,1\}}$ Prover

Prover $\longrightarrow$ Verifier:

$$\begin{cases} \tau & \text{if } b=0 \\ \pi \cdot \tau^{-1} & \text{if } b=1 \end{cases}$$

0) Efficiency
1) Completeness
2) Soundness
3) Perfect Zero-Knowledge

Q0 G'' is isomorphic to G

Q1 G'' is isomorphic to G'

Simulator(G, G')

sample $b \leftarrow \{0,1\}$

if $b=0$	if $b=1$
sample τ	sample τ'
let $G'' = \tau \circ G$	let $G'' = (\tau')^{-1} \circ G'$
output view	output
(G'', b, τ)	$(G'', b, \pi^{-1} \cdot \tau')$

We have perfect ZKP protocol for QR, graph isomorphism

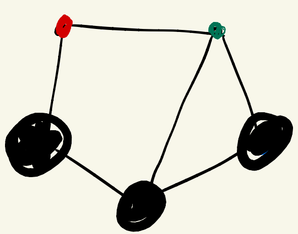
Do we have perfect ZKP for any NP language? **NO!**

— — computational — — — — — — — ? **Yes!** assume OWF

ZKP for NPC language

$G \in$ 3colorable $\iff f: V \to \{0,1,2\}$

s.t. $\forall (u,v) \in E$, $f(u) \neq f(v)$



G, f

Prover

sample $\pi: \{0,1,2\} \to \{0,1,2\}$

G

Verifier

$\pi \circ f$ $\xrightarrow{\text{commit}((\pi \circ f)(v)) \text{ for all } v \in V}$

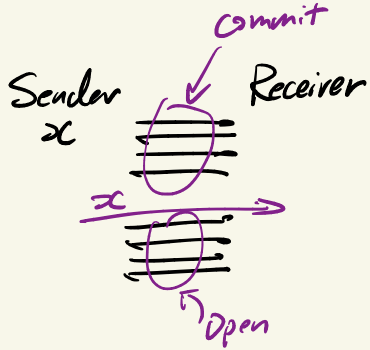
$(u,v) \leftarrow E$

$\xleftarrow{\hspace{3cm}}$

open $((\pi \circ f)(u))$ open $((\pi \circ f)(v))$ $\xrightarrow{\hspace{3cm}}$

Commitment

- commit
- open



Hiding : Commitment reveals no info about x

Computational/Statistical/Perfect

Binding : A commitment cannot be opened to multiple msgs

E.g. let f is an injective OWF

h is a hard-core predicate of f

Commit (b)

Sample random r

$\xrightarrow{f(r),\ h(r)\oplus b}$

Open

$\xrightarrow{r,\ b}$

Computational Hiding

&

Perfect Binding

Malicious Verifier

Zero-Knowledge. $\forall$ p.p.t. V^* $\exists$ p.p.t Simulator S $\forall x \in L$

$$\text{View}_V\left(P(x,w), V^*\right) \approx S(x)$$

e.g. $P^{g,a}$ V^{g}

$h = g \cdot r^2$ $\longrightarrow$

$\longleftarrow$ $b \in \{0,1\}$

$ra^{1-b} = \sqrt{h \cdot g^{-b}}$ $\longrightarrow$

Hybrid $S(g,a)$

Sample b'

$h = g r^2$ $\longrightarrow$

$\longleftarrow$ $b \in \{0,1\}$

if $b = b'$ if $b \neq b'$ rewind

$\Downarrow$

ra^{1-b} $\longrightarrow$

$S(g)$ rewind

$b' \xleftarrow{\$} \{0,1\}$

$h = \begin{cases} g \cdot r^2 & \text{if } b' = 1 \\ s^2 & \text{if } b' = 0 \end{cases}$ $\longrightarrow$

$\longleftarrow$ b

if $b = b'$ if $b \neq b'$ rewind

$\begin{cases} r & b = 1 \\ s & b = 0 \end{cases}$ $\longrightarrow$

Our ZKP for QR is malicious-verifier perfect ZK.

G, f

Prover

sample $\pi: \{0,1,2\} \to \{0,1,2\}$

$\pi \circ f$

commit$((\pi \circ f)(v))$ for all $v \in V$

G

Verifier

$(u,v) \leftarrow ??$

open$((\pi \circ f)(u))$ open$((\pi \circ f)(v))$

The protocol is malicious-verifier computational ZK.

$S(G,f)$

random $(u',v') \in E$

$\pi \circ f$

commit$((\pi \circ f)(u))$ $\forall u \in V$

rewind.

$(u,v) \leftarrow ??$

if $(u,v) = (u',v')$

o.w.

open$((\pi \circ f)(u))$ open$((\pi \circ f)(v))$

$S(G)$

random $(u',v') \in E$

random $f: V \to \{0,1,2\}$ s.t. $f(u') \neq f(v')$

rewind

commit$(f(v))$ $\forall v \in V$

$(u,v) \leftarrow$

if $(u,v) = (u',v')$

if $(u,v) \neq (u',v')$

open$(f(u))$ open$(f(v))$

$P \rightleftarrows V$

3-msg ZKP protocol

amplify soundness

Idea 1
sequential repeat

Idea 2
parallel repeat

Parallel repetition of Malicious-Verifier ZKP protocol is not necessarily malicious-verifier ZK

Malicious Verifier ZKP

Parallel Repetition of ZKP for QR

P

$$\xrightarrow{\begin{array}{c} h_1 = g \cdot r_1^2 \\ h_2 = g \cdot r_2^2 \\ \vdots \\ h_n = g \cdot r_n^2 \end{array}}$$

$$\xleftarrow{b_1 \cdots b_n} \quad V$$

$$\longrightarrow$$

$$r_i a^{1-b} = \begin{cases} r_i \cdot a & b_i = 0 \\ r_i & b_i = 1 \end{cases}$$

Simulator

guess $b_1' \cdots b_n'$

$$h_i = \begin{cases} g \cdot r_i^2 & \text{if } b_i' = 0 \\ s_i^2 & \text{if } b_i' = 1 \end{cases} \longrightarrow$$

$$\xleftarrow{b_1 \cdots b_n}$$

if $(b_1 \cdots b_n) = (b_1' \cdots b_n')$, o.w. rewind

$$\xrightarrow{\begin{cases} r_i \\ s_i \end{cases} \text{ for each } i}$$

Non-interactive ZK

$$\overset{x,w}{P} \xrightarrow{\text{proof } \pi} \overset{x}{V}$$

$$\left.\begin{array}{l}\text{completeness}\\ \text{soundness}\\ \text{zero-knowledge}\end{array}\right\} \Longrightarrow L \in BPP$$

in plain model

Solution I: Random Oracle

$$P \qquad\qquad\qquad V$$

$$\xrightarrow{m_1}$$

$$\xrightarrow{m_2 = RO(m_1, x, id)}$$

$$\xrightarrow{m_3}$$

$$\pi = (m_1, m_2, m_3)$$

Fiat-Shamir

Solution 2: Common Random String (CRS) model
Reference

QNR

$\mathbb{Z}_n$ $(n=pq)$, g

Prove $g \in QNR_N$

$\begin{cases} g \bmod p \text{ is } QNR_p \\ g \bmod q \text{ is } QNR_q \end{cases}$

$CRS = (r_1 \; r_2 \; \cdots \; r_{2n})$

$\Downarrow$

$\boxed{\text{filter}}$ $\left(\frac{r_i}{N}\right) = 1$

$\Downarrow$

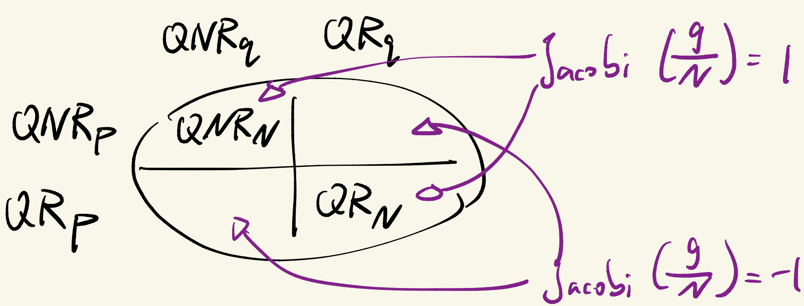
$(r_1, r_2, \ldots, r_n) \xleftarrow{\$} QR_N \cup QNR_N$

$P \xrightarrow[\sqrt{r_i} \text{ or } \sqrt{g \cdot r_i}]{\text{for all } i} V$

Zero-Knowledge

$(crs, \pi) \approx S(g)$

FACT



$g \in QNR$, $\quad r \in QR_N \cup QNR_N$

$\Rightarrow r \in QR_N$ or $g \cdot r \in QR_N$

$g \in QR$ $\quad r \in QNR_N$

$\Rightarrow r \notin QR_N$, $g \cdot r \notin QR_N$

3SAT

$$\begin{tikzcd}[column sep=huge]
& \mathrm{crs} \arrow[dl, bend right] \arrow[dr, bend left] & \\
\begin{matrix} f, x \\ P \end{matrix} \arrow[rr, "N,\ g,\ \pi", "\text{proving } g \in QNR_N"'] \arrow[rr, shift right=6ex, "y_1 \cdots\cdots y_n"] & & \begin{matrix} f \\ V \end{matrix}
\end{tikzcd}$$

$f(x_1 \cdots x_n) =$

$(x_1 \vee x_2 \vee x_3) \wedge$

$(x_2 \vee x_3 \vee \overline{x_4}) \wedge \cdots$

$$\begin{cases} y_i \leftarrow QNR_N & \text{if } x_i = 1 \\ y_i \leftarrow QR_N & \text{if } x_i = 0 \end{cases}$$

$(x_1 \vee x_2 \vee x_3)$

$y_1 \in QNR_N$ or $y_2 \in QNR_N$ or $y_3 \in QNR_N$

ZK Proof → prove "$x \in L$"

ZK Proof of Knowledge → prove "I know witness w"

$\exists$ Extractor Ext $\forall P^*$, if $\Pr[(P^*, V(x)) \to 1] \geq 90\%$

$\mathrm{Ext}(P^*) \to w$ s.t. $M(x, w) = 1$