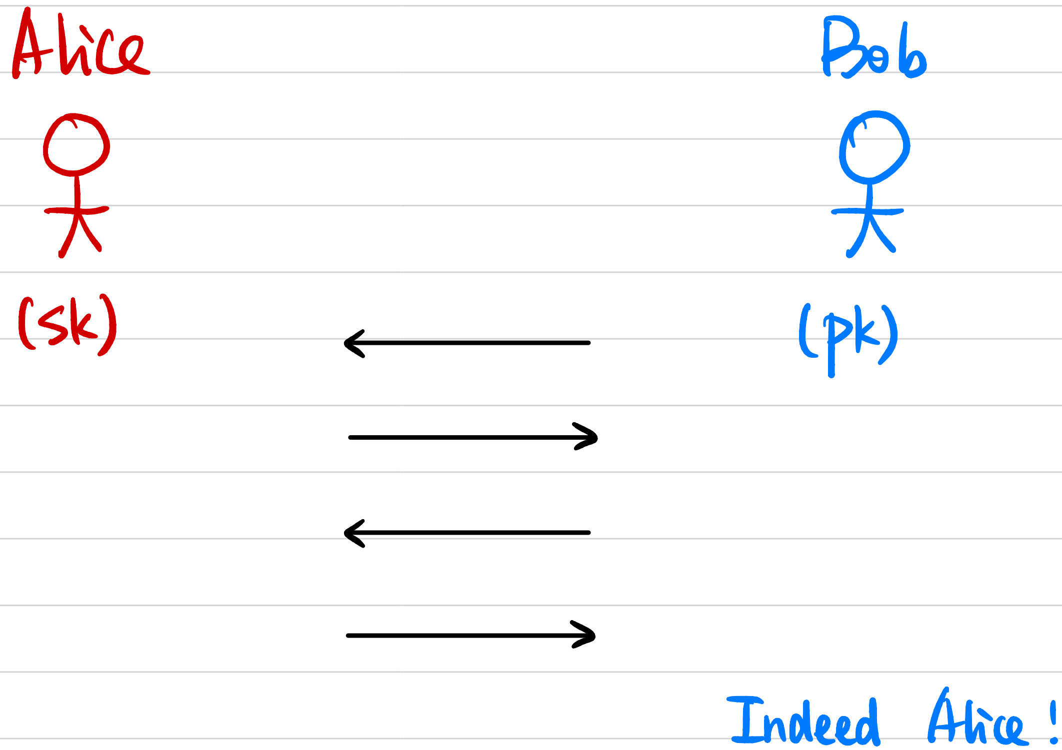
CSCI 1510

- Identification Schemes
- Fiat-Shamir Transform
- Schnorr's Identification / Signature Schemes
- Definition of Zero-Knowledge Proofs
- Perfect ZKP for Diffie-Hellman Tuples

Signatures from DLOG

$$\begin{tikzcd}
\text{1 Identification Scheme} \arrow[r, Rightarrow, "\text{2 Fiat-Shamir}", "\text{Transform}"'] & \text{Digital Signature} \\
\text{Discrete Log} \arrow[u, Rightarrow, "\text{3}", "\text{Schnorr}"']
\end{tikzcd}$$

Identification Scheme



Special 3-Round Identification Scheme

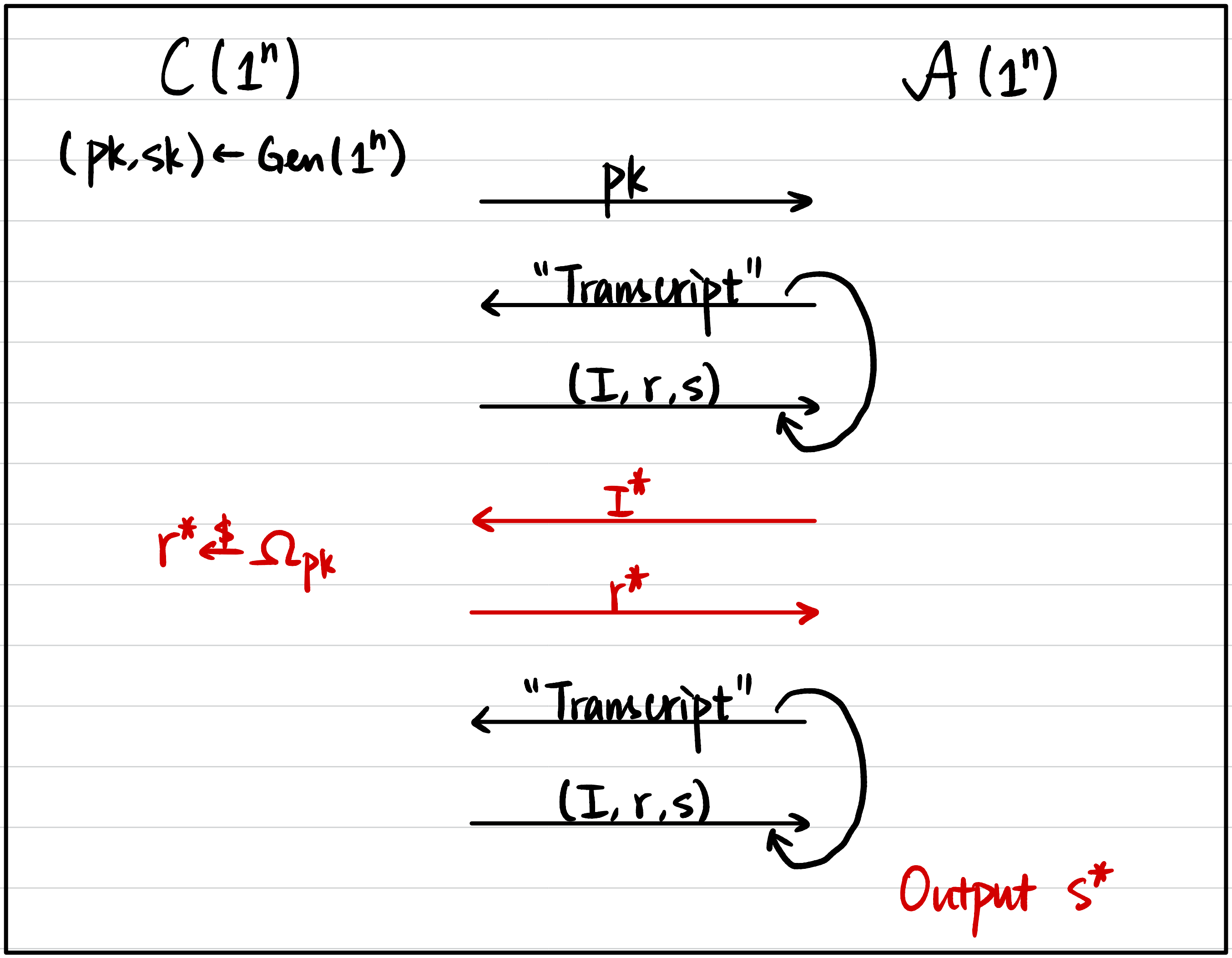
$$(pk, sk) \leftarrow \mathrm{Gen}(1^n)$$

Prover (sk)		Verifier (pk)
$(I, st) \leftarrow P_1(sk)$	$\xrightarrow{\ I\ }$	
		$r \xleftarrow{\$} \Omega_{pk}$
	$\xleftarrow{\ r\ }$	
$s := P_2(sk, st, r)$	$\xrightarrow{\ s\ }$	
		$V(pk, r, s) \stackrel{?}{=} I$

Correctness: If both parties follow the protocol description, then the verifier accepts with probability 1.

Special 3-Round Identification Scheme

Def A 3-round identification scheme $\Pi = (\mathrm{Gen}, P_1, P_2, V)$ is secure if $\forall$ PPT $\mathcal{A}$, $\exists$ negligible function $\varepsilon(\cdot)$ s.t. $\Pr[V(pk, r^*, s^*) = I^*] \le \varepsilon(n)$.



Fiat-Shamir Transform

Let $\Pi = (\mathrm{Gen}_{ID}, P_1, P_2, V)$ be a secure identification scheme.

Construct a signature scheme $\Pi' = (\mathrm{Gen}, \mathrm{Sign}, \mathrm{Vrfy})$:

- $\mathrm{Gen}(1^n)$:

  $(pk, sk) \leftarrow \mathrm{Gen}_{ID}(1^n)$

  Specify a hash function $H: \{0,1\}^* \to \Omega_{pk}$

- $\mathrm{Sign}_{sk}(m)$: $m \in \{0,1\}^*$

  $(I, st) \leftarrow P_1(sk)$

  $r := H(I \| m)$

  $s := P_2(sk, st, r)$

  Output $\sigma = (r, s)$

- $\mathrm{Vrfy}_{pk}(m, \sigma)$:

  $I := V(pk, r, s)$

  Output 1 iff $H(I \| m) = r$.

Thm If Π is secure and H is modeled as a random oracle, then Π' is secure.

Proof Sketch

Π (ID scheme) | Π' (Signature)

$\mathcal{C}$ | $\mathcal{B}$ | $\mathcal{A}$

$\mathcal{C} \xrightarrow{pk} \mathcal{B}$

$i \leftarrow \{1, 2, \cdots, q\}$

$\mathcal{B} \xrightarrow{pk} \mathcal{A}$

$\mathcal{A} \xrightarrow{H \text{ of } x} \mathcal{B}$

Parse $x = I \| m$

If i-th query:

$\mathcal{B} \xrightarrow{m^* = m} \mathcal{C}$

$\mathcal{C} \xrightarrow{r^*} \mathcal{B}$

$H(I\|m) := r^*$

Otherwise: $r \xleftarrow{\$} \Omega_{pk}$, $H(I\|m) := r$

$\mathcal{B} \xrightarrow{H(I\|m)} \mathcal{A}$

$\mathcal{A} \xrightarrow{\text{Signature of } m} \mathcal{B}$

$\mathcal{B} \xrightarrow{\text{"Transcript"}} \mathcal{C}$

$\mathcal{C} \xrightarrow{(I, r, s)} \mathcal{B}$

$H(I\|m) := r$

② $H(I\|m)$ has been queried by $\mathcal{A}$ with negligible probability

$\mathcal{B} \xrightarrow{(r, s)} \mathcal{A}$

$\mathcal{A} \xrightarrow{\text{Output } (m^*, \sigma^*)} \mathcal{B}$

$\sigma^* = (r^*, s^*)$

Output s^*

① $\mathcal{A}$ has queried $H(I^* \| m^*)$ with non-negligible probability

Schnorr's Identification Scheme

$$(\mathbb{G}, q, g) \leftarrow \mathcal{G}(1^n)$$
$$x \xleftarrow{\$} \mathbb{Z}_q$$
$$h := g^x$$

Prover		Verifier
(x)		$(\mathbb{G}, q, g, h)$
$y \xleftarrow{\$} \mathbb{Z}_q$		
$I := g^y$	$\xrightarrow{\quad I \quad}$	
		$r \xleftarrow{\$} \mathbb{Z}_q$
	$\xleftarrow{\quad r \quad}$	
$s := (x \cdot r + y) \bmod q$	$\xrightarrow{\quad s \quad}$	
		$g^s / h^r \stackrel{?}{=} I$

Thm If DLOG is hard relative to $\mathcal{G}$, then this is a secure identification scheme.

Proof Sketch

C — DLOG — B — ID Scheme — A

C $\xrightarrow{(G, q, g, h)}$ B

B $\xrightarrow{(G, q, g, h)}$ A

B $\xleftarrow{\text{"Transcript"}}$ A

$s \xleftarrow{\$} \mathbb{Z}_q$

$r \xleftarrow{\$} \mathbb{Z}_q$

$I := g^s / h^r$

B $\xrightarrow{(I, r, s)}$ A

B $\xleftarrow{I^*}$ A

B $\xrightarrow{r'}$ A

$r^* \xleftarrow{\$} \mathbb{Z}_q$

B $\xrightarrow{r^*}$ A

Rewind

$r' \xleftarrow{\$} \mathbb{Z}_q$

B $\xleftarrow{\text{"Transcript"}}$ A

B $\xrightarrow{(I, r, s)}$ A

B $\xleftarrow{\text{Output } s^*}$ A

B $\xleftarrow{\text{Output } s'}$ A

$$I^* \cdot h^{r^*} = g^{s^*}$$

$$I^* \cdot h^{r'} = g^{s'}$$

$$\Rightarrow h^{r^* - r'} = g^{s^* - s'}$$

$$h = g^{(s^* - s')(r^* - r')^{-1}}$$

Zero-Knowledge Proof (ZKP)

Alice

Bob

[Coke & Pepsi taste differently]

[There is a bug in your code]

[I have the secret key for this ciphertext]

What is a proof?

What does zero-knowledge mean?

Coke & Pepsi

Alice

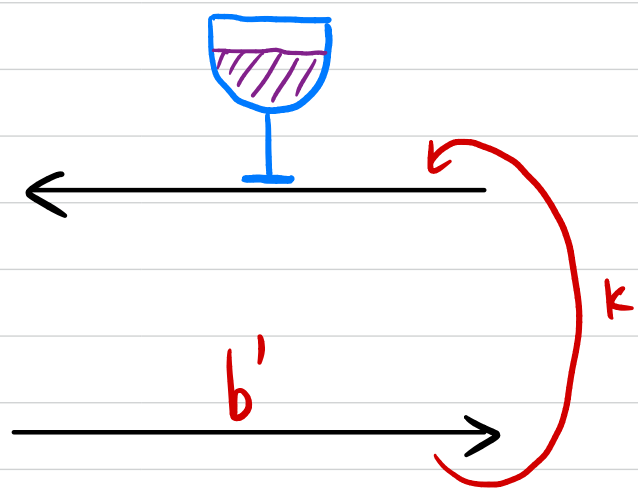
Bob

[Coke & Pepsi taste differently]

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

$b=0$, Coke

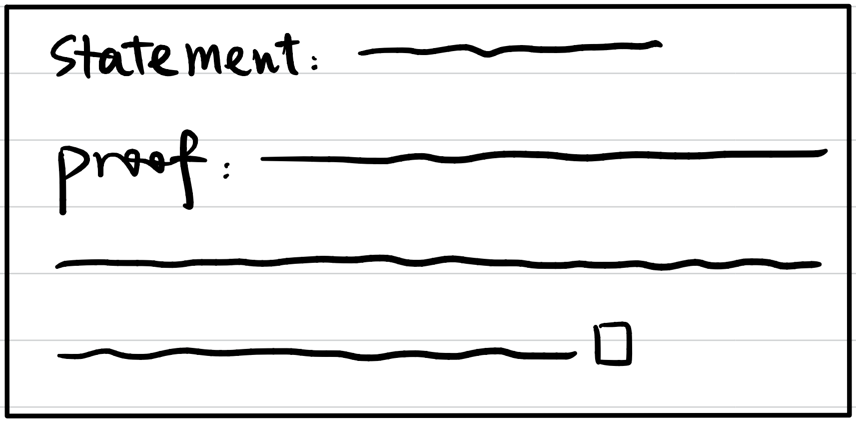
$b=1$, Pepsi



If statement is true: $\Pr[b=b'] = 1$

If statement is false: $\Pr[b=b'] = (1/2)^k$

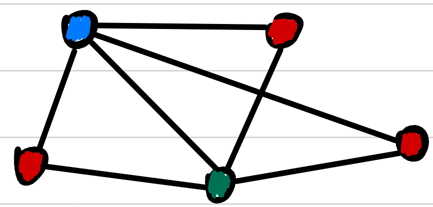
What is a "proof system"?



- **Completeness:** If statement is true, then $\exists$ proof that proves it's true.
- **Soundness:** If statement is false, then $\forall$ proof can't prove it's true.

NP as a Proof System

Example: Gragh 3-Coloring



NP language $L = \{ G : G \text{ has 3-Coloring} \}$

NP relation $R_L = \{ (G, 3COL) \}$

graph 3-coloring

Statement: graph G

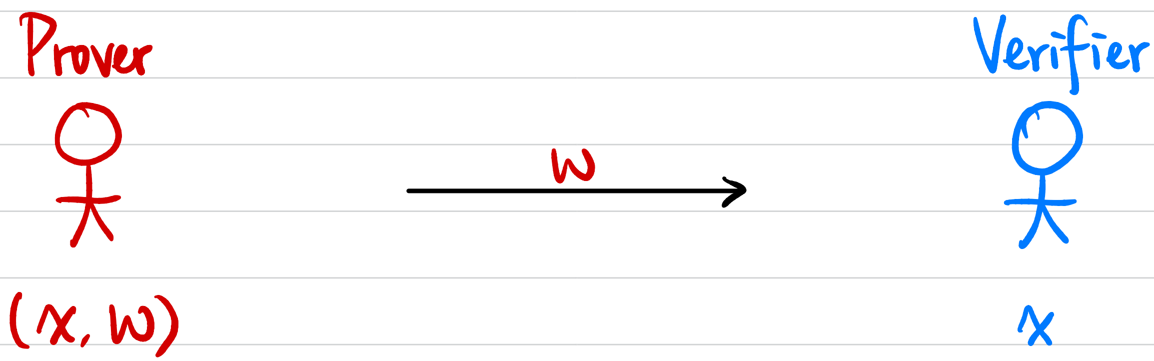
Proof: 3-coloring of G: $3COL$

$(G, 3COL) \in R_L$

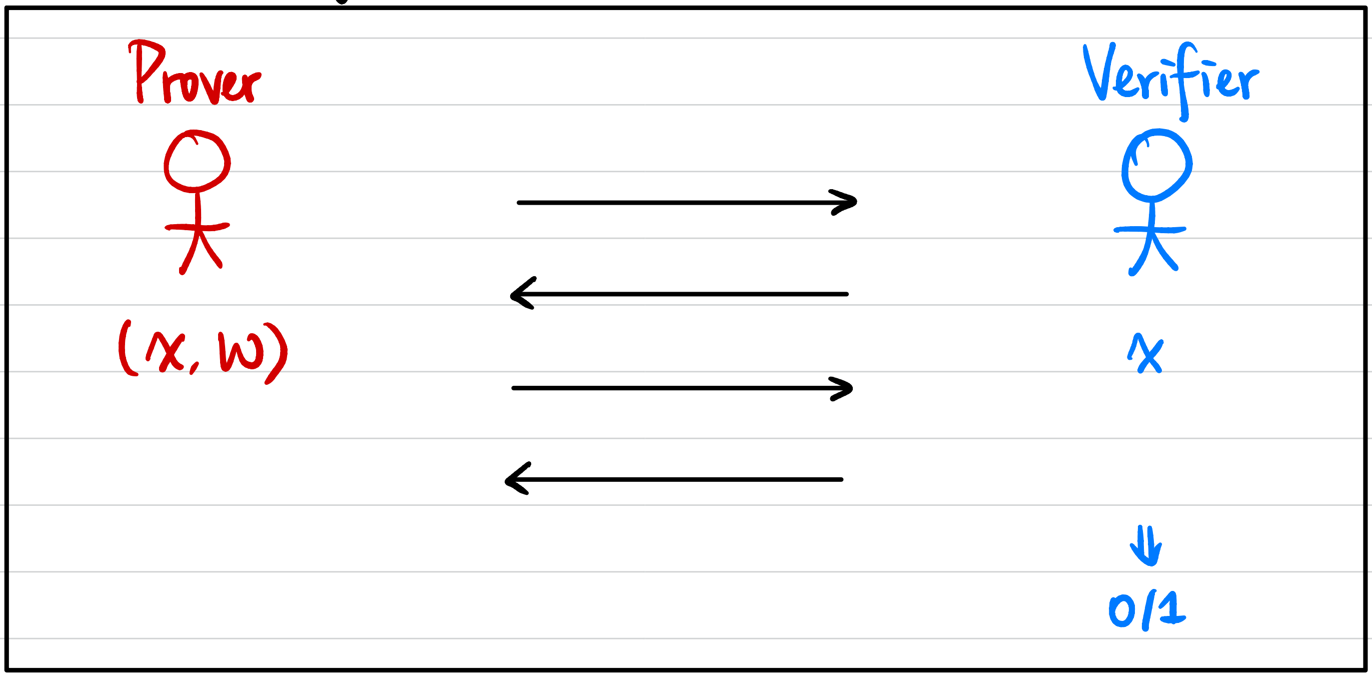
NP as a Proof System

A language L is in NP if $\exists$ poly-time V s.t.

- **Completeness:** $\forall x \in L$, $\exists w$ s.t. $V(x, w) = 1$
  witness
- **Soundness:** $\forall x \notin L$, $\forall w^*$, $V(x, w^*) = 0$



Zero-Knowledge Proof (ZKP)



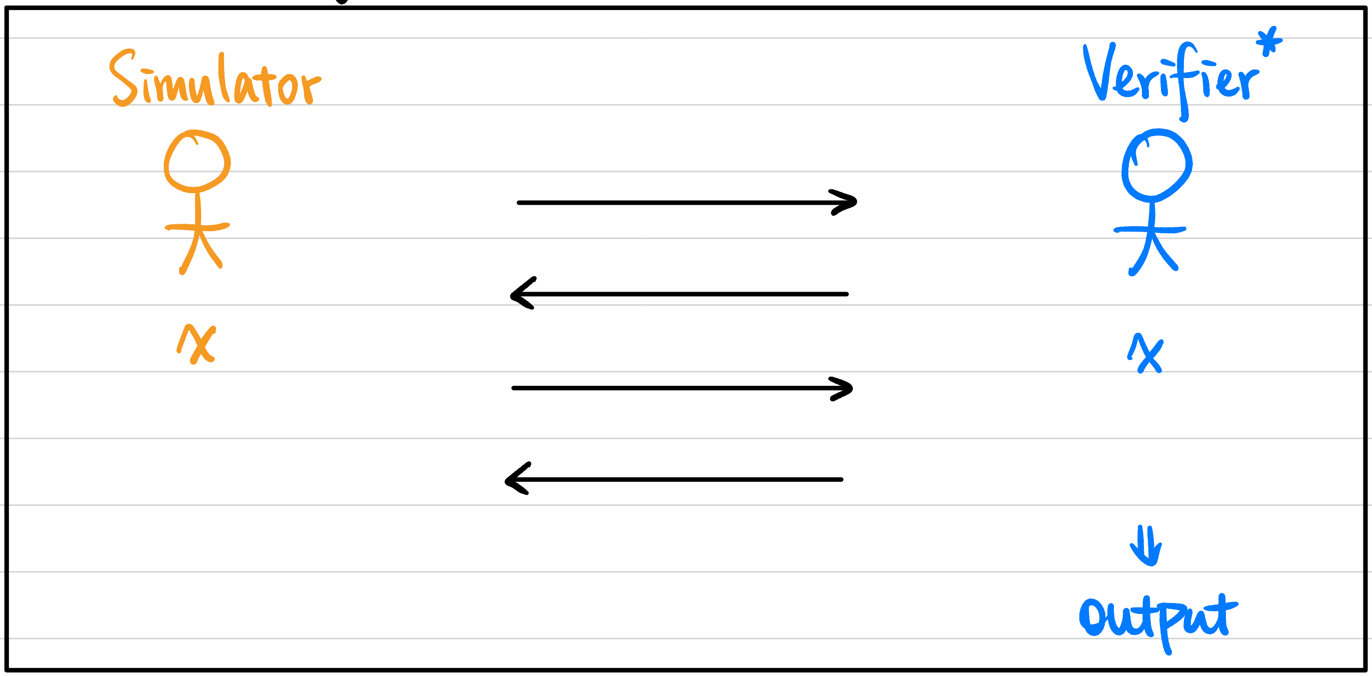
Let (P, V) be a pair of PPT interactive machines. (P, V) is a zero-knowledge proof system for a language L with associated relation R_L if

- Completeness: $\forall (x,w) \in R_L$, $\Pr[\, P(x,w) \longleftrightarrow V(x)$ outputs $1\,] = 1$.
- Soundness: $\forall x \notin L$, $\forall$ (PPT) P^*, $\Pr[\, P^*(x) \longleftrightarrow V(x)$ outputs $1\,] \le \text{negl}(n)$.

  argument

- Zero-Knowledge?

Zero-Knowledge Proof (ZKP)



- **Zero-Knowledge:** $\forall$ PPT V^*, $\exists$ PPT S s.t. $\forall (x, w) \in R_L$,

$$\text{Output}_{V^*}[P(x,w) \longleftrightarrow V^*(x)] \simeq S(x)$$

perfect / statistical / computational

$\equiv$ $\quad\quad$ $\overset{s}{\simeq}$ $\quad\quad$ $\overset{c}{\simeq}$

Perfect ZKP for Diffie-Hellman Tuples

Input: Cyclic group G of order q, generator g, h, u, v

$h = g^a$, $u = g^b$, $v = g^{ab}$

Witness: b

Statement: $\exists b \in \mathbb{Z}_q$ s.t. $u = g^b \wedge v = h^b$

Prover (b)		Verifier
$r \xleftarrow{\$} \mathbb{Z}_q$		
	$\xrightarrow{A := g^r,\ B := h^r}$	
		$\sigma \xleftarrow{\$} \{0,1\}$
	$\xleftarrow{\ \sigma\ }$	
	$\xrightarrow{s := \sigma \cdot b + r \bmod q}$	Verify $g^s = A \cdot u^\sigma$, $h^s = B \cdot v^\sigma$

Completeness?

If $\sigma = 0 \Rightarrow s = r \Rightarrow g^s = A \quad h^s = B$

If $\sigma = 1 \Rightarrow s = b + r \Rightarrow g^s = A \cdot u \quad h^s = B \cdot v$

Soundness?

$(g, h, u, v) \notin L$ $\quad$ $b \neq b'$

$h = g^a, \quad u = g^b, \quad v = g^c = h^{b'}$

$$\forall x \notin L, \ \forall P^*, \quad \Pr[P^*(x) \longleftrightarrow V(x) \text{ outputs } 1] \leq \text{negl}(n)$$

Prover*		**Verifier**
$r \xleftarrow{\$} \mathbb{Z}_q$	$\xrightarrow{A := g^r, \ B := h^{r'}}$	n times
	$\xleftarrow{\sigma}$	$\sigma \xleftarrow{\$} \{0,1\}$
	$\xrightarrow{s \quad (\cancel{S := \sigma \cdot b + r \bmod q})}$	Verify $g^s = A \cdot u^\sigma$, $h^s = B \cdot v^\sigma$

Soundness error 2^{-n}.

$$g^s = A \cdot u^\sigma \Leftrightarrow g^s = g^r \cdot (g^b)^\sigma = g^{r + b\cdot\sigma} \Leftrightarrow s = r + b \cdot \sigma \mod q$$

$$h^s = B \cdot v^\sigma \Leftrightarrow h^s = h^{r'} \cdot (h^{b'})^\sigma = h^{r' + b'\cdot\sigma} \Leftrightarrow s = r' + b' \cdot \sigma \mod q$$

$$r - r' = (b - b') \cdot \sigma$$

If $r = r'$ $\Rightarrow$ caught by V if $\sigma = 1$

If $r \neq r'$ $\Rightarrow$ caught by V if $\sigma = 0$.

Zero-Knowledge?

$\forall$ PPT V^*, $\exists$ PPT S s.t. $\forall (x, w) \in R_L$,

$$\text{Output}_{V^*}\left[P(x,w) \longleftrightarrow V^*(x)\right] \equiv S(x)$$

Simulator		**Verifier***
$r \xleftarrow{\$} \mathbb{Z}_q$	
$A := g^r,\ B := h^r \longrightarrow$
	$\sigma \xleftarrow{\$} \{0,1\}$
$\xleftarrow{\quad \sigma \quad}$
$s := \sigma \cdot b + r \bmod q \longrightarrow$