Scribes?

OT preprocessing + extension

Malicious security (2-party setting)

WI, 2k, POK

The GMW compiler (2-party setting)

Preprocessing

\[ P_1 \xleftarrow{\$} x_0, x \xrightarrow{\text{cOT}} b, x_b \xrightarrow{\$} P_2 \]

\[ s_0, s_1 \xrightarrow{\text{b+c}} c \]

\[ c_0 = s_0 \oplus x_b \oplus c \quad c_0, c_1 \quad s_c = c_c \oplus x_b \]

\[ c_1 = s_1 \oplus x_1 \oplus k \]

\[ = (s_c \oplus x_c \oplus k_c) \oplus x_b \]

\[ = s_c \oplus x_b \]

\[ = s_c \]

OT extension

Parties will generate \( m \) OTs on \( k \)-bit strings from \( K \) OTs on \( m \)-bit strings — base OTs

\[ m \gg k \quad k \sim \text{security parameter} \]

Total cost = \( O(k) \) public key operations + \( O(m) \) private key operations
Aside, given $k$-bit OT, easy to get $m$-bit OT for any $m > k$

\[ \text{Verbal description of the diagram} \]

**Diagram Description**

- **$P_1$**
  - $X_{i_0}$
  - $X_{i_1}$
  - $\vdots$
  - $X_{n_0}$
  - $X_{n_1}$
  - $K$

- **$P_2$**
  - $r = r_i \cdots r_m$

- **Base OT**
  - $s_i^r$
  - $t_i^r \otimes r$

- **$g_1$**
  - $f_i \otimes s_i \cdot r$
  - $f_i \otimes s_i \cdot r$

- **$g_m$**
  - $f_i \otimes s_i \cdot r$

- **$T_i = g_i \otimes c_i \cdot s$**

- **$X_{i_0} \oplus H(g_i)$**
  - $C_{i_0}$

- **$X_{i_1} \oplus H(g_i)$**
  - $C_{i_1}$

- **$x_i \oplus c_i \cdot H(T_i)$**
  - $C_{i_0}$
  - $C_{i_1}$
Malicious security (2-party)

- Real-world execution of protocol \( T \) wi some adversary \( A \)
  
  (Output of honest party, view of \( A \))

- Ideal-world evaluation \( \phi \)

\[
\begin{array}{c}
\text{Input} \quad x \quad \text{Output} \quad y \\
\text{Protocol} \quad \Phi \quad (x, y) \\
\text{Randomness} \quad z_1, z_2 \\
\text{Advantage} \quad 0, 1 \\
\end{array}
\]

Zero-knowledge (proofs of knowledge)

NP language \( L \)

\( \text{i.e., there exists an efficient } R \text{ s.t. } x \in L \iff \exists w \text{ s.t. } R(x,w) = 1 \)

\( \text{i.e., } L = \text{SAT} = \{ \text{Boolean formulas } \phi \text{ that are satisfiable} \} \)

\( R_{\text{SAT}}(\phi, x) = 1 \iff \phi(x) = \text{true} \)

\( \text{i.e., } L = \text{HAM} = \{ \text{directed graph } G \text{ s.t. } G \text{ has a Hamiltonian cycle} \} \)

\( R_{\text{HAM}}(G, v_1, v_2, \ldots, v_n) = 1 \iff v_1, \ldots, v_n \text{ is a Hamiltonian cycle in } G \)

\[
\begin{array}{c}
\text{Input} \quad x, w \\
\text{Randomness} \quad z_2 \text{, } k \text{, } z_1 \\
\text{Advantage} \quad 0, 1 \\
\end{array}
\]

\( b = 1 \iff x' = x \quad \text{and } R_L(x', w) = 1 \)
Zero-Knowledge proofs - evaluating $\mathcal{D}_{zk}$ against a malicious $V$
proofs of Knowledge - evaluating $\mathcal{D}_{zk}$ against a malicious $P$

ZK POK - securely evaluating $\mathcal{D}_{zk}$

Show ZK POK protocol for an $\text{NP}$-complete language

$\Rightarrow$ ZK POK protocol for all of $\text{NP}$

Let $G$ be a graph with $n$ vertices and $m$ edges.

1. Choose a random permutation $\pi$ of $\{0, 1, \ldots, n-1\}$
2. Let $G'$ be the graph obtained by applying $\pi$ to $G$.
3. If $b = 0$, prove that $A$ corresponds to $G$.
4. If $b = 1$, prove that $A$ has an empty Hamiltonian cycle.
5. If $b = 0$, open everything and show isomorphism to $G$.
6. If $b = 1$, open Hamiltonian cycle only and verify.