Exercise 5

Zero-Knowledge Proofs

Loïc Baccigalupi

October 21, 2022

5.2 The Permuted Kernel Problem

i) For the protocol to be perfect complete, we need the Verify calls to be always equal to 1. This implies that $Q \mid\mid HQ^{-1}\mathbf{w}$ and $R \mid\mid \mathbf{w} - bR\mathbf{v}$ need to be the correct messages for the commitments and openings (A, d_A) and (B, d_B) respectively, i.e.:

$$\text{Commit}(pp, Q \mid\mid HQ^{-1}\mathbf{w}) = (A, d_A)$$
$$\text{Commit}(pp, R \mid\mid \mathbf{w} - bR\mathbf{v}) = (B, d_B)$$

Because (A, d_A) and (B, d_B) are computed at the start of the protocol, the two messages need to be computable from the start. We set $m_A = Q \mid\mid HQ^{-1}\mathbf{w}$ and $m_B = R \mid\mid \mathbf{w} - bR\mathbf{v}$.

We can fill in the first part as follows:

- Generate $X \in \mathbb{Z}_2^{N\times N}$ and $\mathbf{r} \in \mathbb{Z}_2^N$ uniformly at random.
- Set $m_A = X \mid\mid HX^{-1}\mathbf{rv}$ and $m_B = XHP \mid\mid \mathbf{rv}$.
- Set $Q = X$ and $R = XHP$.
- Generate $\text{Commit}(pp, m_A) = (A, d_A)$ and $\text{Commit}(pp, m_B) = (B, d_B)$.
- Peggy has now computed A and B.

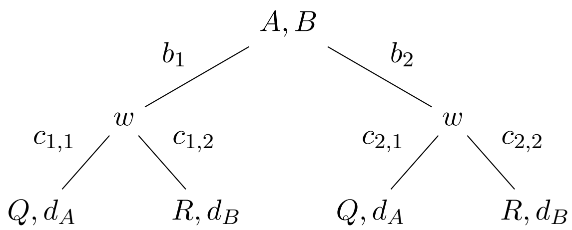
In the second part, we set $\mathbf{w} = \mathbf{rv}$.

We now have:

$$Q \mid\mid HQ^{-1}\mathbf{w} = X \mid\mid HX^{-1}\mathbf{rv} = m_A$$
$$R \mid\mid \mathbf{w} - bR\mathbf{v} = XHP \mid\mid \mathbf{rv} - bXHPv = XHP \mid\mid \mathbf{rv} = m_B$$

Which proves perfect correctness.

The protocol is also (2,2)-special-sound. The tree of accepting transcript is:



Because at each branch, $c_{i,1} \neq c_{i,2}$, the know that one leaf should have Q, d_A and the other one R, d_B (in the diagram, $c_{i,1} = 0$ and $c_{i,2} = 1$ without loss of generality).
The extractor E has then access to Q and R and can compute:

$$H^{-1}Q^{-1}R = H^{-1}X^{-1}XHP = H^{-1}HP = P$$

And succesfully extract the witness.

ii) Proof of special honest-verifier zero-knowledge:

1) What is the verifier's view ?

The verifier's view is: (A, B, c, S, d_S), where $S \in \{A, B\}$ and $d_S \in \{d_A, d_B\}$.

2) What does the simulator do ?

- If $c = 0$:
  - Generate $Q \in \mathbb{Z}_2^{N \times N}$ and $\mathbf{r} \in \mathbb{F}_2^N$ uniformly at random.
  - Set $\mathbf{w} = \mathbf{rv}$.
  - Compute $(A, d_A) = \text{Commit}(pp, Q \mid\mid HQ^{-1}\mathbf{rv})$.
  - Generate $B \in \mathcal{C}$ uniformly at random.
- If $c = 1$:
  - Generate $R \in \mathbb{Z}_2^{N \times N}$ and $\mathbf{r} \in \mathbb{F}_2^N$ uniformly at random.
  - Set $\mathbf{w} = \mathbf{rv}$.
  - Compute $(B, d_B) = \text{Commit}(pp, R \mid\mid \mathbf{rv})$.
  - Generate $A \in \mathcal{C}$ uniformly at random.

iii)