More is Merrier: Relax the Non-Collusion Assumption in Multi-Server PIR

Tiantian Gong, Ryan Henry, Alexandros Psomas, Aniket Kate

What is Private Information Retrieval (PIR)? [CKGS95]

A \((k, t)\)-PIR scheme for a database \(D\) with \(n\) entries:

- \(k\)-out-of-\(k\), \(t\)-private
- \(0 \leq t < k\)

\[ q_1, q_2, \ldots, q_k \] is the query index set:

- \(q_1 \in \{0,1\}^n\)

\[ \exists \ y \in [n] \text{ s.t. } \sum_{i=1}^{k} q_i[y] = 1 \]

\[ D = \{ \mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_n \} \]

\[ \mathcal{D}_y = \{ \mathcal{D}_{y_i} \} \]

Answer string \(a_1, a_2, \ldots, a_k\)

- \(a_i\) is the value of \(D_i\) at index \(y_i\)

Security

- Correctness – C can reconstruct \(D\) from \(\sum_{i=1}^{k} q_i \cdot a_i = q\)
- Privacy (IT, computational) – less than \(n + 1\) parties learn extra info:
  \[ H(X|Y) = H(X) \]

\(H()\) computes the entropy of a random variable; \(X\) is the random variable for \(x\)

1. Multi-Server 1-Private PIR is Efficient

Our model: \((\ell, k, 1)\)-PIR where \(\ell \geq k \geq 2\), database size \(N = n \cdot k\)
- \(\ell\) total servers, \(k\)-out-of-\(k\), 1-private

Communication complexity

\begin{align*}
\text{Single-server} & \quad \text{polylogarithmic} \\
2\text{-Server} & \quad \frac{\mathcal{O}(\log n \cdot \log \log n)}{N} \quad [DG16] \\
3\text{-Server} & \quad \frac{2^{\mathcal{O}(\log n \cdot \log \log n)}}{N} \quad [Efremenko09] \\
k\text{-Server} & \quad \frac{\mathcal{O}(\log^3 n \cdot \log \log n)}{N} \quad \text{[CKGS95]} \quad \text{[MMW23]} \quad \text{VS} \quad \frac{N}{q} \quad \text{bit operations per server} \quad [HH19]
\end{align*}

2. What we do and How

What: Relax the non-collusion assumption to rationality assumption, i.e., servers are rational or malicious

How: Given two premises / our facts

1. We cannot directly detect collusion and collusion can be realized with any protocol, e.g., MPC, TEE, magical clouds, etc.
2. After successful collusion, at least some colluding parties have learned something nontrivial about the index \(x\) or entry \(D_x\) -- denote as \(f(x)\)

3. Mechanism overview

Mechanism \(M\)

- Unknown: secrets \(f(x), f(x')\)
- Winner selection rule \(W\)
- Payment rule \(P\)

4. Protocol overview

Assume a secure commitment scheme:
- Commit(\(x\))
- \(\ell\)-way OR
- \(\ell\)-way AND

5. Communication and computation overhead

On paper

- One additional commitment per message – instantiated with SHA-3 implementation as a smart contract on Ethereum
- CheckCircuits(\(x\)) checks if the function is trivial with oracle services

Table 1. Gas costs summary