Quantum Private Information Retrieval

Private Information Retrieval with Entangled Servers

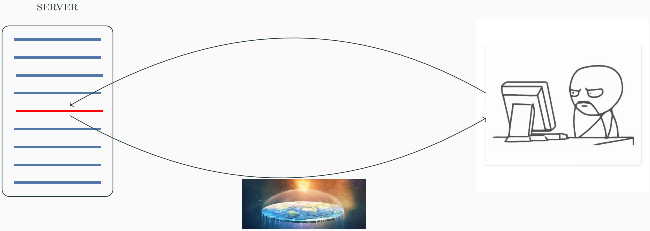
Tefjol Pllaha

Joint with M. Allaix, L. Holzbaur, and C. Hollanti

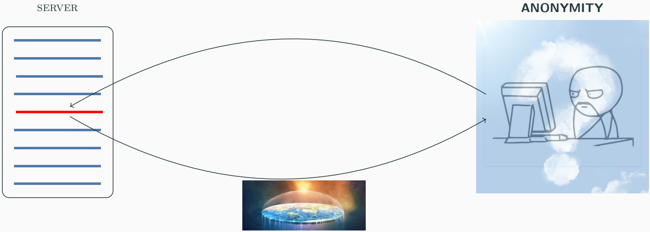
Department of Communications and Networking

Aalto University, Finland

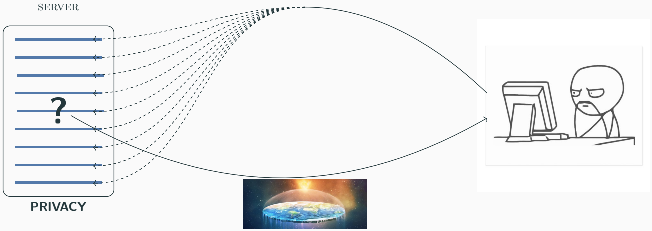
Online Privacy



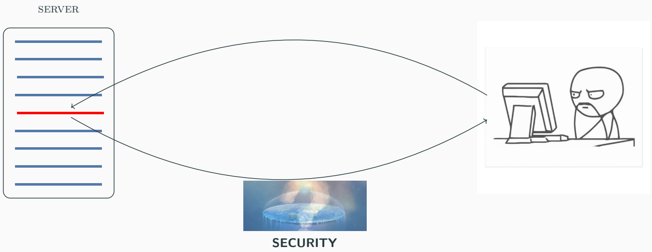
Online Privacy



Online Privacy



Online Privacy



Some Milestones

- Chor et al. 1995: Seminal paper on PIR.
  - Privacy **only** achievable by downloading the entire database.
  - PIR schemes for replicated databases.
  - **Impractical due to storage overhead.**

Some Milestones

- Chor et al. 1995: Seminal paper on PIR.
  - Privacy **only** achievable by downloading the entire database.
  - PIR schemes for replicated databases.
  - **Impractical due to storage overhead.**
- Renewed interest from coded storages.
  - Collusion, capacity, lower overhead...

Some Milestones

- Chor et al. 1995: Seminal paper on PIR.
  - Privacy **only** achievable by downloading the entire database.
  - PIR schemes for replicated databases.
  - **Impractical due to storage overhead.**
- Renewed interest from coded storages.
  - Collusion, capacity, lower overhead...
- Increased demand/awareness for privacy.
  - Anonymization, differential privacy, data protection laws ...

Some Milestones

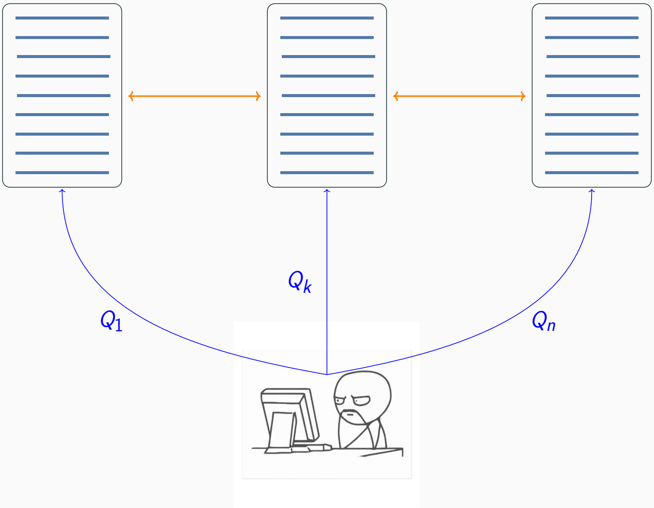
- Chor et al. 1995: Seminal paper on PIR.
  - Privacy **only** achievable by downloading the entire database.
  - PIR schemes for replicated databases.
  - **Impractical due to storage overhead.**
- Renewed interest from coded storages.
  - Collusion, capacity, lower overhead...
- Increased demand/awareness for privacy.
  - Anonymization, differential privacy, data protection laws ...
- Quest for practical solutions continues.

Coded Storage

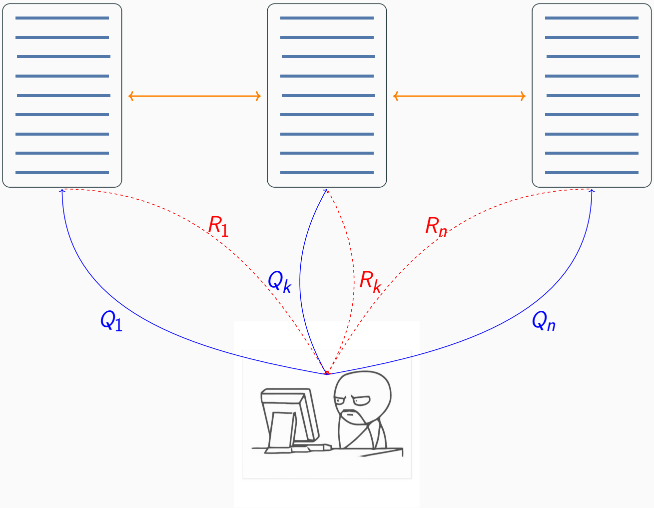
m files $x^1, \ldots, x^m \in \mathbb{F}_q^{\beta \times k}$ are encoded and stored on n servers by a $[n, k]$ storage code $\mathcal{C}$.

$$\begin{array}{r} \\ \text{file } 1 \\ \\ \\ \\ \\ \text{file } m \\ \\ \end{array} \begin{pmatrix} x^1_{1,1} & \cdots & x^1_{1,k} \\ \vdots & \ddots & \vdots \\ x^1_{\beta,1} & \cdots & x^1_{\beta,k} \\ \vdots & \vdots & \vdots \\ x^m_{1,1} & \cdots & x^m_{1,k} \\ \vdots & \ddots & \vdots \\ x^m_{\beta,1} & \cdots & x^m_{\beta,k} \end{pmatrix} \cdot \mathbf{G}_{\mathcal{C}} = \begin{array}{c} \begin{pmatrix} y^1_{1,1} & \cdots & y^1_{1,n} \\ \vdots & \ddots & \vdots \\ y^1_{\beta,1} & \cdots & y^1_{\beta,n} \\ \vdots & \vdots & \vdots \\ y^m_{1,1} & \cdots & y^m_{1,n} \\ \vdots & \ddots & \vdots \\ y^m_{\beta,1} & \cdots & y^m_{\beta,n} \end{pmatrix} \\ \begin{array}{ccc} \mathrm{SERVER}_1 & & \mathrm{SERVER}_n \end{array} \end{array}$$

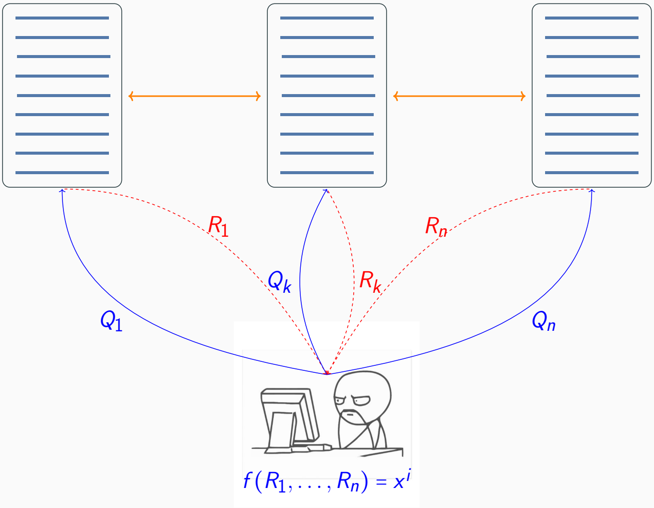
Private Information Retrieval (PIR)



Private Information Retrieval (PIR)



Private Information Retrieval (PIR)



PIR with t-collusion (t-PIR)

Definition (t-PIR).

User privacy: Any set of at most t colluding nodes learns no information about the index i of the desired file, *i.e.*, the mutual information

$$I(i; Q_{\mathcal{T}}^K, R_{\mathcal{T}}^K, y_{\mathcal{T}}) = 0, \quad \forall\ \mathcal{T} \subset [n], |\mathcal{T}| \leq t\ .$$

Server privacy: The user does not learn any information about the files other than the requested one, *i.e.*,

$$I(x^j; Q^K, R^K, K) = 0, \quad \forall j \neq K\ .$$

A scheme with both user and server privacy is called *symmetric*.

PIR with t-collusion (t-PIR)

Definition (Rate and Capacity).

For a PIR scheme the **rate** is the number of information bits of the requested file retrieved per downloaded bits, *i.e.*,

$$R_{\text{PIR}} = \frac{\text{Number of bits in a file}}{\text{Number of downloaded bits}} .$$

The PIR **capacity** is the supremum of PIR rates of all possible PIR schemes, for a fixed parameter setting.

PIR with t-collusion (t-PIR)

Definition (Rate and Capacity).

For a PIR scheme the rate is the number of information bits of the requested file retrieved per downloaded bits, *i.e.*,

$$R_{\mathrm{PIR}} = \frac{\text{Number of bits in a file}}{\text{Number of downloaded bits}} \,.$$

The PIR capacity is the supremum of PIR rates of all possible PIR schemes, for a fixed parameter setting.

Convention

QPIR is PIR with "entangled servers".

PIR with t-collusion (t-PIR)

Definition (Rate and Capacity).

For a PIR scheme the rate is the number of information bits of the requested file retrieved per downloaded bits, *i.e.*,

$$R_{\mathrm{PIR}} = \frac{\text{Number of bits in a file}}{\text{Number of downloaded bits}}\ .$$

The PIR capacity is the supremum of PIR rates of all possible PIR schemes, for a fixed parameter setting.

Convention

QPIR is PIR with "entangled servers".

Motivated by the work of Seunghoan Song and Masahito Hayashi

- arXiv:2001.04436, arXiv:1903.12556, arXiv:1903.10209
- Replicated storage with $t = n - 1$ collusion.

PIR with t-collusion (t-PIR)

Definition (Rate and Capacity).

For a PIR scheme the rate is the number of information bits of the requested file retrieved per downloaded bits, *i.e.*,

$$R_{\text{PIR}} = \frac{\text{Number of bits in a file}}{\text{Number of downloaded bits}} .$$

The PIR capacity is the supremum of PIR rates of all possible PIR schemes, for a fixed parameter setting.

Convention

QPIR is PIR with "entangled servers".

Motivated by the work of Seunghoan Song and Masahito Hayashi

- arXiv:2001.04436, arXiv:1903.12556, arXiv:1903.10209
- Replicated storage with $t = n - 1$ collusion.
- **Goal**: $[n, k]$ coded storage with $t = n - k$ collusion.

Ingredients for QPIR

- **Star Product** PIR scheme from Freij-Hollanti et al.
  - Coded storage with storage code $\mathcal{C}$.
  - A **retrieval** code $\mathcal{D}$ that determines the privacy.
  - Scheme with **rate** $(d_{\mathcal{C}\star\mathcal{D}}-1)/n$ that protects against $d_{\mathcal{D}^\perp}-1$ **collusions**.

Ingredients for QPIR

- Star Product PIR scheme from Freij-Hollanti et al.
  - Coded storage with storage code $\mathcal{C}$.
  - A retrieval code $\mathcal{D}$ that determines the privacy.
  - Scheme with rate $(d_{\mathcal{C}\star\mathcal{D}} - 1)/n$ that protects against $d_{\mathcal{D}^\perp} - 1$ collusions.
  - $d_{\mathcal{C}_1\star\mathcal{C}_2} - 1 \leq \max\{0, n - (\dim(\mathcal{C}_1) + \dim(\mathcal{C}_2) - 1)\}$.

Ingredients for QPIR

- Star Product PIR scheme from Freij-Hollanti et al.
  - Coded storage with storage code $\mathcal{C}$.
  - A retrieval code $\mathcal{D}$ that determines the privacy.
  - Scheme with rate $(d_{\mathcal{C}\star\mathcal{D}} - 1)/n$ that protects against $d_{\mathcal{D}^\perp} - 1$ collusions.
  - $d_{\mathcal{C}_1\star\mathcal{C}_2} - 1 \leq \max\{0, n - (\dim(\mathcal{C}_1) + \dim(\mathcal{C}_2) - 1)\}$.
- **Generalized Reed-Solomon codes**

$$\mathrm{GRS}_k(\alpha, v) = \{(v_i f(\alpha_i))_{1\leq i\leq n} \mid f(x) \in \mathbb{F}_q^{<k}[x]\}.$$

Ingredients for QPIR

- Star Product PIR scheme from Freij-Hollanti et al.
  - Coded storage with storage code $\mathcal{C}$.
  - A retrieval code $\mathcal{D}$ that determines the privacy.
  - Scheme with rate $(d_{\mathcal{C}\star\mathcal{D}} - 1)/n$ that protects against $d_{\mathcal{D}^\perp} - 1$ collusions.
  - $d_{\mathcal{C}_1\star\mathcal{C}_2} - 1 \leq \max\{0, n - (\dim(\mathcal{C}_1) + \dim(\mathcal{C}_2) - 1)\}$.
- Generalized Reed-Solomon codes

$$\mathrm{GRS}_k(\alpha, v) = \{(v_i f(\alpha_i))_{1\leq i\leq n} \mid f(x) \in \mathbb{F}_q^{<k}[x]\}.$$

- **Quantum Computation**.

Ingredients for QPIR: Quantum Computation

- **Bell State** $|\Phi\rangle = (|00\rangle + |11\rangle)/\sqrt{2}$.

Ingredients for QPIR: Quantum Computation

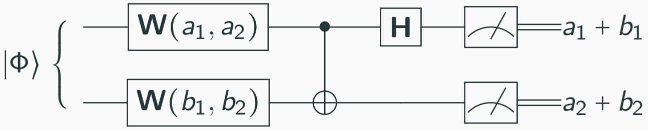
- Bell State $|\Phi\rangle = (|00\rangle + |11\rangle)/\sqrt{2}$.
- **Weyl Operator** $\mathbf{W}(a, b) = \mathbf{X}^a \mathbf{Z}^b$.

Ingredients for QPIR: Quantum Computation

- Bell State $|\Phi\rangle = (|00\rangle + |11\rangle)/\sqrt{2}$.
- Weyl Operator $\mathbf{W}(a, b) = \mathbf{X}^a\mathbf{Z}^b$.
- **The PVM**
  $\mathcal{B}_{\mathbb{F}_2^2} = \{\mathbf{B}_{(a,b)} = \mathbf{W}_1(a, b)|\Phi\rangle\langle\Phi|\mathbf{W}_1(a, b)^{\mathrm{t}} \mid a, b \in \mathbb{F}_2\}$.

Ingredients for QPIR: Quantum Computation

- Bell State $|\Phi\rangle = (|00\rangle + |11\rangle)/\sqrt{2}$.
- Weyl Operator $\mathbf{W}(a,b) = \mathbf{X}^a \mathbf{Z}^b$.
- The PVM
  $\mathcal{B}_{\mathbb{F}_2^2} = \{\mathbf{B}_{(a,b)} = \mathbf{W}_1(a,b)|\Phi\rangle\langle\Phi|\mathbf{W}_1(a,b)^{\mathrm{t}} \mid a,b \in \mathbb{F}_2\}$.
- **Two-Sum Protocol**: Alice and Bob send the sum (a_1+b_1, a_2+b_2) of their bits to Carol.



A QPIR Example

- $n = 4$ servers and $[4,2]_4$ - coded database with RS code

$$\mathbf{G}_{\mathcal{C}} = \begin{pmatrix} 1 & 0 & \alpha^2 & \alpha \\ 0 & 1 & \alpha & \alpha^2 \end{pmatrix}.$$

A QPIR Example

- $n = 4$ servers and $[4, 2]_4$ - coded database with RS code

$$\mathbf{G}_{\mathcal{C}} = \begin{pmatrix} 1 & 0 & \alpha^2 & \alpha \\ 0 & 1 & \alpha & \alpha^2 \end{pmatrix}.$$

- **Files**: m files in $x^i \in \mathbb{F}_4^{\beta \times k}$
  - $\beta = 1$ and $k = 2$ (determined by encoding).
  - $x^i = (x^i_1, x^i_2)$.
  - k also determines the number of rounds.

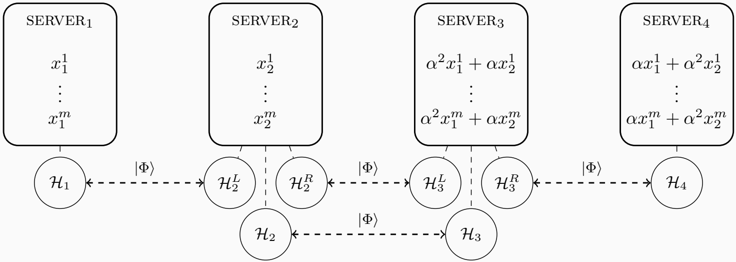
A QPIR Example

- $n = 4$ servers and $[4, 2]_4$ - coded database with RS code

$$\mathbf{G}_{\mathcal{C}} = \begin{pmatrix} 1 & 0 & \alpha^2 & \alpha \\ 0 & 1 & \alpha & \alpha^2 \end{pmatrix}.$$

- **Files**: m files in $x^i \in \mathbb{F}_4^{\beta \times k}$
  - $\beta = 1$ and $k = 2$ (determined by encoding).
  - $x^i = (x_1^i, x_2^i)$.
  - k also determines the number of rounds.
- Query index K, i.e., the requested file is $x^K = (x_1^K, x_2^K)$.

A QPIR Example: Entangled Servers



A QPIR Example: Queries

- Generate two independent and uniformly random vectors $Z_1, Z_2 \in \mathbb{F}_4^m$.

A QPIR Example: Queries

- Generate two independent and uniformly random vectors $Z_1, Z_2 \in \mathbb{F}_4^m$.
- Encode Z_1, Z_2 as codewords of the **dual** code:

$$\begin{aligned}(Q_1, Q_2, Q_3, Q_4) &= (Z_1, Z_2) \cdot \mathbf{G}_{\mathcal{C}^\perp} + \xi_{K,1} \\ &= [Z_1, Z_2, \alpha^2 Z_1 + \alpha Z_2, \alpha Z_1 + \alpha^2 Z_2^2] + \xi_{K,1}.\end{aligned}$$

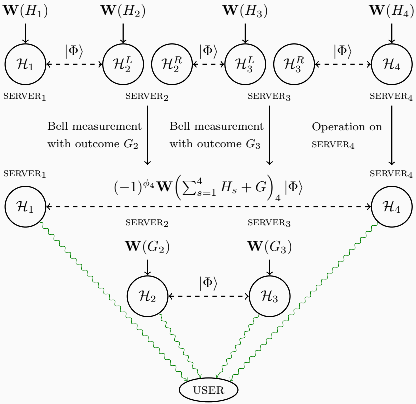
A QPIR Example: Queries

- Generate two independent and uniformly random vectors $Z_1, Z_2 \in \mathbb{F}_4^m$.
- Encode Z_1, Z_2 as codewords of the **dual** code:

$$\begin{aligned}(Q_1, Q_2, Q_3, Q_4) &= (Z_1, Z_2) \cdot \mathbf{G}_{\mathcal{C}^\perp} + \xi_{K,1} \\ &= [Z_1, Z_2, \alpha^2 Z_1 + \alpha Z_2, \alpha Z_1 + \alpha^2 Z_2^2] + \xi_{K,1}.\end{aligned}$$

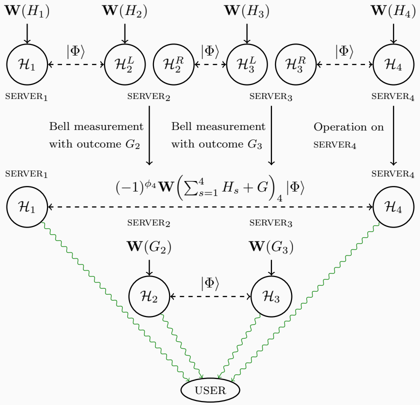
- Query Q_s to server s.

A QPIR Example: Servers' Response



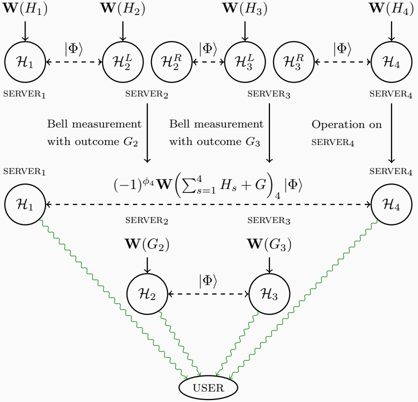
- Each server computes $H_s = \langle\, Q_s \,|\, y_s \,\rangle \in \mathbb{F}_4 = \mathbb{F}_2^2$.

A QPIR Example: Servers' Response



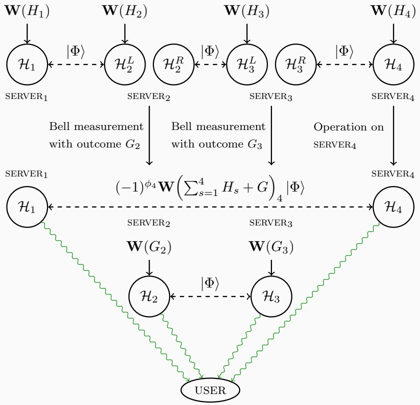
- Each server computes $H_s = \langle\, Q_s \,|\, y_s \,\rangle \in \mathbb{F}_4 = \mathbb{F}_2^2$.
- Servers 1,4: $\mathbf{W}(H_1)$, $\mathbf{W}(H_4)$ to $\mathcal{H}_1$, $\mathcal{H}_4$, respectively.

A QPIR Example: Servers' Response



- Each server computes $H_s = \langle\, Q_s \,|\, y_s \,\rangle \in \mathbb{F}_4 = \mathbb{F}_2^2$.
- Servers 1,4: $\mathbf{W}(H_1)$, $\mathbf{W}(H_4)$ to $\mathcal{H}_1$, $\mathcal{H}_4$, respectively.
- Servers 2,3: $\mathbf{W}(H_s)$ to $\mathcal{H}_s^L$, Bell measurement on $\mathcal{H}_s^L \otimes \mathcal{H}_s^R$ with outcome $G_s \in \mathbb{F}_2^2$, $\mathbf{W}(G_s)$ to $\mathcal{H}_s$.

A QPIR Example: Servers' Response



- Each server computes $H_s = \langle\, Q_s \,|\, y_s \,\rangle \in \mathbb{F}_4 = \mathbb{F}_2^2$.
- Servers 1,4: $\mathbf{W}(H_1)$, $\mathbf{W}(H_4)$ to $\mathcal{H}_1$, $\mathcal{H}_4$, respectively.
- Servers 2,3: $\mathbf{W}(H_s)$ to $\mathcal{H}_s^L$, Bell measurement on $\mathcal{H}_s^L \otimes \mathcal{H}_s^R$ with outcome $G_s \in \mathbb{F}_2^2$, $\mathbf{W}(G_s)$ to $\mathcal{H}_s$.
- Each server sends its qubit to the user.

A QPIR Example: Retrieval

- Measure $\mathcal{H}_2 \otimes \mathcal{H}_3$ to retrieve $G = G_2 + G_3$ (two-sum protocol).

A QPIR Example: Retrieval

- Measure $\mathcal{H}_2 \otimes \mathcal{H}_3$ to retrieve $G = G_2 + G_3$ (two-sum protocol).
- Apply $\mathbf{W}(G)$ to $\mathcal{H}_4$ and measure to retrieve x_1^K.

A QPIR Example: Retrieval

- Measure $\mathcal{H}_2 \otimes \mathcal{H}_3$ to retrieve $G = G_2 + G_3$ (two-sum protocol).
- Apply $\mathbf{W}(G)$ to $\mathcal{H}_4$ and measure to retrieve x_1^K.
- Repeat everything to retrieve x_2^K and build the **desired** file $x_2^K = (x_1^K, x_2^K)$.

A QPIR Example: Retrieval

- Measure $\mathcal{H}_2 \otimes \mathcal{H}_3$ to retrieve $G = G_2 + G_3$ (two-sum protocol).
- Apply $\mathbf{W}(G)$ to $\mathcal{H}_4$ and measure to retrieve x_1^K.
- Repeat everything to retrieve x_2^K and build the **desired** file $x_2^K = (x_1^K, x_2^K)$.

Remark

Here we targeted servers 1&2 (systematic encoding). Since the storage is MDS-coded, one can target any two (k in general) servers.

A QPIR Example: Secrecy, Collusion, Rate

- **User secrecy**: queries $Q_1, \ldots, Q_4$ independent of the index K, two random vectors generated and encoded into queries $\Rightarrow$ at least three servers needed in order to retrieve the file requested $\Rightarrow$ **2-collusion**.

A QPIR Example: Secrecy, Collusion, Rate

- User secrecy: queries $Q_1, \ldots, Q_4$ independent of the index K, two random vectors generated and encoded into queries $\Rightarrow$ at least three servers needed in order to retrieve the file requested $\Rightarrow$ 2-collusion.

- **Server secrecy**: obtained for any p because the received state of the user is independent of the fragments x_p^i with $i \neq K$ and the measurement outcome $G^{(p)}$ is independent of any file.

A QPIR Example: Secrecy, Collusion, Rate

- User secrecy: queries $Q_1, \dots, Q_4$ independent of the index K, two random vectors generated and encoded into queries $\Rightarrow$ at least three servers needed in order to retrieve the file requested $\Rightarrow$ 2-collusion.

- Server secrecy: obtained for any p because the received state of the user is independent of the fragments x_p^i with $i \neq K$ and the measurement outcome $G^{(p)}$ is independent of any file.

- **Rate**: $R = \frac{2 \cdot 2}{2 \cdot 4} = \frac{1}{2}$.

QPIR with n Servers

- Base field: $\mathbb{F}_{4^L}$ where $L = \min\{\ell \mid 4^\ell \geq n\}$.

QPIR with n Servers

- Base field: $\mathbb{F}_{4^L}$ where $L = \min\{\ell \mid 4^\ell \geq n\}$.
- Files: $\mathcal{X} = \left\{x_b^i \in \mathbb{F}_{4^L}^k \mid i \in [m], b \in [\beta]\right\}$

QPIR with n Servers

- Base field: $\mathbb{F}_{4^L}$ where $L = \min\{\ell \mid 4^\ell \geq n\}$.
- Files: $\mathcal{X} = \left\{x_b^i \in \mathbb{F}_{4^L}^k \mid i \in [m], b \in [\beta]\right\}$
  - Stripes $x_b^i = (x_{b,1}^i, \ldots, x_{b,k}^i)$.

QPIR with n Servers

- Base field: $\mathbb{F}_{4^L}$ where $L = \min\{\ell \mid 4^\ell \geq n\}$.
- Files: $\mathcal{X} = \left\{x_b^i \in \mathbb{F}_{4^L}^k \mid i \in [m], b \in [\beta]\right\}$
    - Stripes $x_b^i = (x_{b,1}^i, \ldots, x_{b,k}^i)$.
    - File size $F = 2kL\beta$.

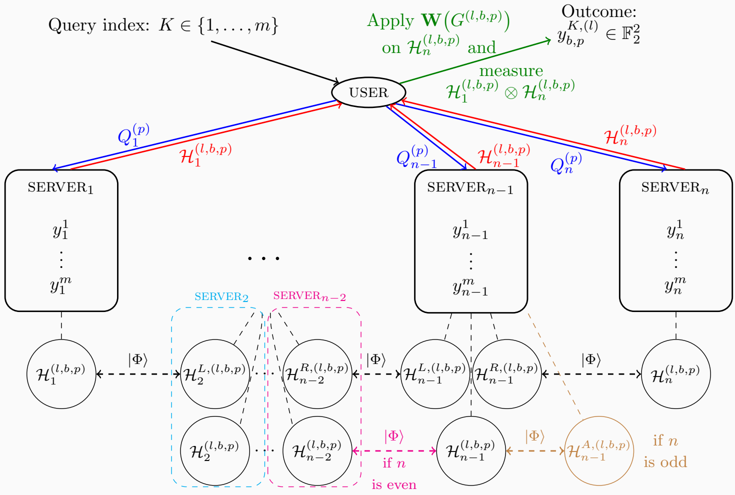
QPIR with n Servers

- Base field: $\mathbb{F}_{4^L}$ where $L = \min\{\ell \mid 4^\ell \geq n\}$.
- Files: $\mathcal{X} = \left\{x_b^i \in \mathbb{F}_{4^L}^k \mid i \in [m], b \in [\beta]\right\}$
  - Stripes $x_b^i = (x_{b,1}^i, \ldots, x_{b,k}^i)$.
  - File size $F = 2kL\beta$.
- Encoding: $\mathcal{C} = \mathrm{GRS}_k(\alpha, \mathbf{1}^k)$.

QPIR with n Servers

- Base field: $\mathbb{F}_{4^L}$ where $L = \min\{\ell \mid 4^\ell \geq n\}$.
- Files: $\mathcal{X} = \left\{x_b^i \in \mathbb{F}_{4^L}^k \mid i \in [m], b \in [\beta]\right\}$
  - Stripes $x_b^i = (x_{b,1}^i, \ldots, x_{b,k}^i)$.
  - File size $F = 2kL\beta$.
- Encoding: $\mathcal{C} = \mathrm{GRS}_k(\alpha, \mathbf{1}^k)$.
- Query index: K.

QPIR with n Servers



QPIR with n Servers

- $H_s, G_s \in \mathbb{F}_{4^L}$: **packetization** in vectors of $(\mathbb{F}_2^2)^L$.

QPIR with n Servers

- $H_s, G_s \in \mathbb{F}_{4^L}$: packetization in vectors of $(\mathbb{F}_2^2)^L$.
- Up to $(n-k)$-**collusion**: generate $t \leq n-k$ random vectors in $(\mathbb{F}_{4^L})^m$, encode them with $\mathcal{C}^\perp$.

QPIR with n Servers

- $H_s, G_s \in \mathbb{F}_{4^L}$: packetization in vectors of $(\mathbb{F}_2^2)^L$.
- Up to $(n-k)$-collusion: generate $t \leq n-k$ random vectors in $(\mathbb{F}_{4^L})^m$, encode them with $\mathcal{C}^\perp$.
- Server secrecy: **symmetric PIR scheme**.

QPIR with n Servers

- $H_s, G_s \in \mathbb{F}_{4^L}$: packetization in vectors of $(\mathbb{F}_2^2)^L$.
- Up to $(n-k)$-collusion: generate $t \leq n-k$ random vectors in $(\mathbb{F}_{4^L})^m$, encode them with $\mathcal{C}^\perp$.
- Server secrecy: symmetric PIR scheme.
- **Upload cost negligible** to the file size.

QPIR with n Servers

- $H_s, G_s \in \mathbb{F}_{4^L}$: packetization in vectors of $(\mathbb{F}_2^2)^L$.
- Up to $(n-k)$-collusion: generate $t \leq n-k$ random vectors in $(\mathbb{F}_{4^L})^m$, encode them with $\mathcal{C}^\perp$.
- Server secrecy: symmetric PIR scheme.
- Upload cost negligible to the file size.
- **Rate**: With $n = k + t$

$$R_{\text{PIR}} = \begin{cases} \frac{2}{n}, & \text{if } n \text{ is even,} \\ \frac{2}{n+1}, & \text{if } n \text{ is odd,} \end{cases}$$

Improvements with Locally Repairable Codes (LRC)

Definition

An $[n, k]$ code $\mathcal{C}$ is said to have (r, ρ)-locality if there exists a partition $\mathcal{P} = \{\mathcal{A}_1, \ldots, \mathcal{A}_\mu\}$ of $[n]$ into sets $\mathcal{A}_l$ with $\mathcal{A}_l \leq r + \rho - 1, \ \forall l \in [\mu]$ such that for the distance of the code restricted to the positions indexed by $\mathcal{A}_l$ it holds that $d(\mathcal{C}_{\mathcal{A}_l}) \geq \rho, \ \forall \ l \in [\mu]$.

Improvements with Locally Repairable Codes (LRC)

Definition

An $[n, k]$ code $\mathcal{C}$ is said to have (r, ρ)-locality if there exists a partition $\mathcal{P} = \{\mathcal{A}_1, ..., \mathcal{A}_\mu\}$ of $[n]$ into sets $\mathcal{A}_l$ with $\mathcal{A}_l \leq r + \rho - 1, \ \forall l \in [\mu]$ such that for the distance of the code restricted to the positions indexed by $\mathcal{A}_l$ it holds that $d(\mathcal{C}_{\mathcal{A}_l}) \geq \rho, \ \forall \ l \in [\mu]$.

Optimal LRC achieve the **Singleton-like bound**

$$d \leq n - k + 1 - \left(\left\lceil \frac{k}{r} \right\rceil - 1\right)(\rho - 1).$$

Improvements with Locally Repairable Codes (LRC)

Definition

An $[n, k]$ code $\mathcal{C}$ is said to have (r, ρ)-locality if there exists a partition $\mathcal{P} = \{\mathcal{A}_1, \ldots, \mathcal{A}_\mu\}$ of $[n]$ into sets $\mathcal{A}_l$ with $\mathcal{A}_l \leq r + \rho - 1, \; \forall l \in [\mu]$ such that for the distance of the code restricted to the positions indexed by $\mathcal{A}_l$ it holds that $d(\mathcal{C}_{\mathcal{A}_l}) \geq \rho, \; \forall \; l \in [\mu]$.

Optimal LRC achieve the Singleton-like bound

$$d \leq n - k + 1 - \left(\left\lceil \frac{k}{r} \right\rceil - 1 \right) (\rho - 1).$$

The local codes $\mathcal{C}_{\mathcal{A}_l}$ of an optimal LRC $\mathcal{C}$ are $[r + \rho - 1, r]$-MDS.

Improvements with Locally Repairable Codes (LRC)

For $t = \rho - 1$, the retrieval rate of LRC-based QPIR scheme is

$$R_{\mathrm{QPIR}} = \begin{cases} \frac{2}{r+t}, & \text{if } r+t \text{ is even,} \\ \frac{2}{r+t+1}, & \text{if } r+t \text{ is odd,} \end{cases}.$$

Improvements with Locally Repairable Codes (LRC)

For $t = \rho - 1$, the retrieval rate of LRC-based QPIR scheme is

$$R_{\mathrm{QPIR}} = \begin{cases} \frac{2}{r+t}, & \text{if } r+t \text{ is even,} \\ \frac{2}{r+t+1}, & \text{if } r+t \text{ is odd,} \end{cases} .$$

- Improved retrieval rate.

Improvements with Locally Repairable Codes (LRC)

For $t = \rho - 1$, the retrieval rate of LRC-based QPIR scheme is

$$R_{\mathrm{QPIR}} = \begin{cases} \frac{2}{r+t}, & \text{if } r+t \text{ is even,} \\ \frac{2}{r+t+1}, & \text{if } r+t \text{ is odd,} \end{cases} .$$

- Improved retrieval rate.
- Trade-off with server collusion/failure.

Improvements with Locally Repairable Codes (LRC)

For $t = \rho - 1$, the retrieval rate of LRC-based QPIR scheme is

$$R_{\text{QPIR}} = \begin{cases} \frac{2}{r+t}, & \text{if } r+t \text{ is even}, \\ \frac{2}{r+t+1}, & \text{if } r+t \text{ is odd}, \end{cases} .$$

- Improved retrieval rate.
- Trade-off with server collusion/failure.
  - $t = \rho - 1$ colluding nodes, provided that no more than t nodes collude per local group.
  - For such collusion patterns, the scheme can resist collusion of up to $t\mu = (\rho - 1)\mu$ servers

Thank You!